**Detailed Design Specifications (DDS)**

**Build Phase**

|  |  |
| --- | --- |
| Project Name | Click here to enter name |
| Release Number | Click here to enter number |
| **PRIME Clarity Project ID** | Click here to enter number |
| Component Name | CIFS Backup |
| Author | Click here and enter name(s) |
| Revision |  |

**Responsibility List**

| Action | Responsibility |
| --- | --- |
| Owner | Development Manager |
| Primary Peer Review | QA Manager |
| Primary Peer Review | Sustaining Engineering Manager |
| Primary Peer Review | Tech Info. Manager |
| Primary Peer Review | Support Delivery Managers |
| Secondary Peer Review | SPT PRJ-Support SME |
| Secondary Peer Review | Localization Delivery Manager |
| Secondary Peer Review | GIS Manager |
| Secondary Peer Review | Product Manager |

**Note:** The Responsibility List reflects those included in the PRIME Clarity peer review process. Primary peer review is mandatory and must be complete for process to continue. Secondary peer review is recommended, but process will continue without it.

The project document deliverables should be stored in the projects’ PRIME Clarity collaboration tab (per the following policies: [WLD - Records and Information Management Program Policies](http://qms.ca.com/documents/default.asp?srchID=9155) and [Control of Source Code and Design Documents Policy](http://qms.ca.com/documents/default.asp?srchID=6483)).

**Change History:**

| **Revision Date** | **Last Revision By** | **Reason for Change** |
| --- | --- | --- |
| 28th March, 2016 | Haiyang.Ling | Draft v0.1  Add sequence diagram and main EWS Api/EWS operations |
| 30th March, 2016 | Haiyang.Ling | Draft v0.2  Modify backup job sequence diagram |
|  |  |  |

[PREFACE 5](#_Toc446959684)

[1 INTroduction 6](#_Toc446959685)

[1.1 Scope 6](#_Toc446959686)

[1.1.1 Requirements 6](#_Toc446959687)

[1.2 Definitions, Acronyms and Abbreviations 6](#_Toc446959688)

[1.3 References 6](#_Toc446959689)

[2 externals 7](#_Toc446959690)

[2.1 Technical Approach User Perspective 7](#_Toc446959691)

[2.2 Detailed Use Cases 7](#_Toc446959692)

[2.3 Administration Perspective 7](#_Toc446959693)

[2.4 Migration issues 7](#_Toc446959694)

[2.5 Security impact 7](#_Toc446959695)

[3 Architecture 8](#_Toc446959696)

[3.1 Overview 8](#_Toc446959697)

[3.2 Sample Flow/Execution Diagram 8](#_Toc446959698)

[3.3 Integration, Interfaces, and Dependencies 12](#_Toc446959699)

[3.3.1 Integration 12](#_Toc446959700)

[3.3.2 Interfaces 14](#_Toc446959701)

[3.3.3 Certifications 15](#_Toc446959702)

[3.3.4 Other Dependencies 15](#_Toc446959703)

[3.4 Non-Functional Feature Design (AKA the abilities) 15](#_Toc446959704)

[3.4.1 ACCESSABILITY 16](#_Toc446959705)

[3.4.2 INTEGRATEABILITY 16](#_Toc446959706)

[3.4.3 Interoperability 16](#_Toc446959707)

[3.4.4 LocalizaBILITY 17](#_Toc446959708)

[3.4.5 Scalability 17](#_Toc446959709)

[3.4.6 Securability 17](#_Toc446959710)

[3.4.7 Supportability 17](#_Toc446959711)

[3.4.8 upgradeability 17](#_Toc446959712)

[3.4.9 USABILITY 18](#_Toc446959713)

[3.4.10 Supported Platforms 18](#_Toc446959714)

[3.4.11 A list of all platform versions/levels that the product/component will operate on is included in the PRD. 18](#_Toc446959715)

[3.5 PRODUCT design and Specifications 18](#_Toc446959716)

[3.5.1 Installation/Uninstall 18](#_Toc446959717)

[3.5.2 Availability 18](#_Toc446959718)

[3.5.3 Migration 18](#_Toc446959719)

[3.5.4 Performance 18](#_Toc446959720)

[3.5.5 Product Licensing Schema 19](#_Toc446959721)

[3.6 Design limitations, assumptions, and issues 19](#_Toc446959722)

[3.6.1 Limitation 19](#_Toc446959723)

[3.6.2 Issues 19](#_Toc446959724)

[4 Internals 20](#_Toc446959725)

[4.1 Programming details 20](#_Toc446959726)

[4.1.1 Module 1 20](#_Toc446959727)

[4.1.2 Module 2 20](#_Toc446959728)

[4.2 Internationalization 21](#_Toc446959729)

[4.3 Localization 21](#_Toc446959730)

[4.4 Updated effort estimate 21](#_Toc446959731)

[5 Impact Summary 22](#_Toc446959732)

[5.1 Product/Component summary 22](#_Toc446959733)

[5.2 Documentation summary 22](#_Toc446959734)

[5.3 Patent information 22](#_Toc446959735)

[6 Quality Issues 23](#_Toc446959736)

[6.1 Testing recommendations and risk assessment 23](#_Toc446959737)

[7 Packaging and Installation Impact 24](#_Toc446959738)

[8 technical Support Impact 25](#_Toc446959739)

# PREFACE

The Detailed Design Specifications (DDS) describes external functional specifications as well as design specifications for one component or design entity of a project. There can be many DDSs for each project, one for each major feature described in the Product Requirements Document (PRD).

This is intended to be a living document. The product development cycle is a dynamic process in which the project and its criteria for success are refined over time. Therefore, it is expected that the completed DDS will undergo many revisions during the course of a project as requirements, resources, and constraints evolve.

The Development Manager is responsible for the contents of this document. Deliverables that must be completed prior to releasing this document are:

• Product Requirements Document (PRD)

• Top Level Design Specification (TLDS)

Notes:

If this is a Component Project and the component is a third-party open-source or off-the-shelf piece, then a DDS is not required. If the component is developed to specification by a third-party, then the Development organization in that third-party’s organization should provide a DDS or equivalent, excluding the sections related to Patent Information and Programming Details, which need to be completed only if the agreement with the third-party includes source code. In this case, the responsibility list should be modified to show the third-party Development organization as the document Owner and the remaining individuals as Peer Reviewers.

All template instructions can be identified by their gray italic type. This information may be removed after completing the necessary project information. If a section is not applicable for a project, identify the section as “N/A.”

# INTroduction

## Scope

Identify the requirement being addressed and the feature being designed.

### Requirements

## Definitions, Acronyms and Abbreviations

Provide the definitions of terms, acronyms and abbreviations required to interpret this document. For consistency, use the format in the following examples:

Design entity: An element (component) of a design that is structurally and functionally distinct from other elements and is separately named and referenced.

Detailed Design Specifications (DDS): A representation of a software system or component of a system created to facilitate analysis, planning, implementation, and decision-making. The DDS is used as the primary medium for communicating software design information.

## References

Provide a list of documents referenced elsewhere in this document.

Identify each document by title, report number (if applicable), date and publishing organization.

Specify the sources from which the references can be obtained.

Click here to begin typing.

# externals

## Technical Approach User Perspective

TODO: Once we have UI design from Michael, we will add it here.

## Detailed Use Cases

## Administration Perspective

Include any special installation and setup tasks, system parameters, or other preparations that are necessary prior to use. Describe the steps needed to set up and get the component going and any ongoing administration that will need to be performed.

For component projects, this section should be used as the administration section of a Developer’s guide targeted to the component stakeholder Development staff.

Click here to add additional definitions

## Migration issues

Describe the steps required to migrate from existing versions of this software to this version, taking into consideration previously translated projects:

* Is the component going to be backward compatible?
* Can this version co-exist with an older version?
* Can this version be upgraded from the English project to the translated project?

For Component Projects, this section should be used as the administration section of a Developer’s guide targeted to the component stakeholder Development staff.

Click here to begin typing.

This is a new feature. No migration process needed.

## Security impact

Does anything about the function need securing? Could it do any damage? Could it cause the display of sensitive information?

Does the implementation methodology do anything that produces a potential security exposure?

Click here to begin typing

# Architecture

## Overview

Describe the architecture of the feature and where it fits in the overall design. Architectural design may be represented in many forms, including text, graphical description, pseudo-code representation, or combination. Where applicable cite areas of the Top Level Design Specification (TLDS) to reduce duplication.

## Sample Flow/Execution Diagram

With what other components does this component need to interact and how do these relationships work? Often, you will be able to depict these relationships graphically using structure charts, data flow diagrams or transaction diagrams. Show a sample execution of features including the flow and interaction of different components involved.

1. Backup job



1. Write an item to original/alternative location in restore job



1. Dump an item with msg/eml/vcf/vcs format in restore job



1. Main EWS API method and EWS Operation
   * 1. [ExchangeService.AutodiscoverUrl](https://msdn.microsoft.com/en-us/library/microsoft.exchange.webservices.data.exchangeservice.autodiscoverurl(v=exchg.80).aspx) method

Description: Initializes the Exchange Web Services (EWS) URL for a specified email address by calling the Autodiscover service. We need get a valid web service url to connect exchange online service.

Ref Url: <https://msdn.microsoft.com/en-us/library/microsoft.exchange.webservices.data.exchangeservice.autodiscoverurl(v=exchg.80).aspx>

* + 1. [ExchangeService.SyncFolderHierarchy method](https://msdn.microsoft.com/en-us/library/microsoft.exchange.webservices.data.exchangeservice.syncfolderhierarchy(v=exchg.80).aspx)

Description: Folder synchronization syncs a folder structure, or folder hierarchy. The synchronization method or operation returns a list of changes.

Ref Url: <https://msdn.microsoft.com/en-us/library/office/dn440952(v=exchg.150).aspx>

* + 1. [ExchangeService.SyncFolderItems method](https://msdn.microsoft.com/en-us/library/microsoft.exchange.webservices.data.exchangeservice.syncfolderitems(v=exchg.80).aspx)

Description: Item synchronization syncs the items within a folder. The synchronization method or operation returns a list of changes.

Ref Url: <https://msdn.microsoft.com/en-us/library/office/dn440952(v=exchg.150).aspx>

* + 1. [Folder.Bind](https://msdn.microsoft.com/en-us/library/microsoft.exchange.webservices.data.folder.bind(v=exchg.80).aspx) method

Description: Binds to a folder. By the method, we can get folder details by folderId which is got from SyncFolderHierarchy.

Ref Url: <https://msdn.microsoft.com/en-us/library/microsoft.exchange.webservices.data.folder.bind(v=exchg.80).aspx>

* + 1. [Folder.Load](https://msdn.microsoft.com/en-us/library/microsoft.exchange.webservices.data.serviceobject.load(v=exchg.80).aspx)method

Description: Loads properties for folders..

Ref Url: <https://msdn.microsoft.com/en-us/library/microsoft.exchange.webservices.data.serviceobject.load(v=exchg.80).aspx>

* + 1. [[Folder.Delete](https://msdn.microsoft.com/en-us/library/microsoft.exchange.webservices.data.folder.delete(v=exchg.80).aspx) method](https://msdn.microsoft.com/en-us/library/microsoft.exchange.webservices.data.exchangeservice.syncfolderhierarchy(v=exchg.80).aspx)

Description: Delete a folder.

Ref Url: <https://msdn.microsoft.com/en-us/library/microsoft.exchange.webservices.data.folder.delete(v=exchg.80).aspx>

* + 1. [[Folder.Save](https://msdn.microsoft.com/en-us/library/microsoft.exchange.webservices.data.folder.save(v=exchg.80).aspx) method](https://msdn.microsoft.com/en-us/library/microsoft.exchange.webservices.data.exchangeservice.syncfolderitems(v=exchg.80).aspx)

Description: Saves the specified folder.

Ref Url: <https://msdn.microsoft.com/en-us/library/microsoft.exchange.webservices.data.folder.save(v=exchg.80).aspx>

* + 1. [Item.Bind](https://msdn.microsoft.com/en-us/library/microsoft.exchange.webservices.data.item.bind(v=exchg.80).aspx) method

Description: Get an item, including more details about item, such as subject, body, created time, eml data etc.

Ref Url: <https://msdn.microsoft.com/en-us/library/microsoft.exchange.webservices.data.item.bind(v=exchg.80).aspx>

* + 1. [[Item.Delete](https://msdn.microsoft.com/en-us/library/office/dd635072(v=exchg.80).aspx) method](https://msdn.microsoft.com/en-us/library/microsoft.exchange.webservices.data.exchangeservice.syncfolderhierarchy(v=exchg.80).aspx)

Description: Delete an item.

Ref Url: <https://msdn.microsoft.com/en-us/library/office/dd635072(v=exchg.80).aspx>

* + 1. [ExportItems](https://msdn.microsoft.com/en-us/library/office/ff709523(v=exchg.150).aspx) EWS operation

Description: Export items with full fidelity. The result file is a stream with bin format. It can be as a parameter when importing item.

Ref Url: [https://msdn.microsoft.com/en-us/library/office/dn672317(v=exchg.150).aspx#bk\_exportfullfidelity](https://msdn.microsoft.com/en-us/library/office/dn672317(v=exchg.150).aspx%23bk_exportfullfidelity)

* + 1. [UploadItems](https://msdn.microsoft.com/en-us/library/office/ff709490(v=exchg.150).aspx) EWS operation

Description: You can use the [UploadItems](https://msdn.microsoft.com/en-us/library/office/ff709490(v=exchg.150).aspx) EWS operation to upload an item as a data stream. This data stream representation of an item has to come from the results of an [ExportItems](https://msdn.microsoft.com/en-us/library/office/ff709523(v=exchg.150).aspx) operation call.

Ref Url: [https://msdn.microsoft.com/en-us/library/office/dn672319(v=exchg.150).aspx#sectionSection1](https://msdn.microsoft.com/en-us/library/office/dn672319(v=exchg.150).aspx%23sectionSection1)

## Integration, Interfaces, and Dependencies

### Integration

*In these sections, define how the product will be integrated with other products and components.*

#### CA Product/Common Component Integration

In the section below, list all of the products developed by CA and components with which the product integrates. For those that are new or have an impact on the design, define design considerations and/or how it will be integrated (consume, supply, or both). Include the version and consider localization requirements. The key decision template in the Appendix can be used to document significant decisions. Once Common Components have been identified, they need to be registered at the [Tech Stack Website](http://techstack.ca.com/).

##### Catalyst/USM:

*If Catalyst is not supported, state why not.*

Click here to begin typing

##### EEM:

*If EEM is not used for Authentication and/or Authorization, state why not.*

Click here to begin typing

##### Business Objects:

*If Business Objects is not used for Reporting, state why not.*

Click here to begin typing

##### Other CA Product/Common Component

*If other CA product/common component is not used for reporting, state why not.*

Click here to begin typing

#### Third-Party Software

The license text of ALL third-party software must be approved by the Worldwide Law department before it can be shipped with a CA product.  The mechanism for doing this is a Third-Party Software Request (TPSR). The TPSR approval process is performed via an automated process on the [Tech Stack Website](http://techstack.ca.com/) and should be completed prior to Phase 1: Business Case Review, with the understanding that all third-party selections may change after Phase 1. This process ensures that TPSRs are recorded, approved by the Worldwide Law department and that royalty payments are made by CA.  The key decision template in the Appendix can be used to document significant decisions.

Please enter information for filed TPSRs below, and notify Tech Info. if there are documentation issues related to any TPSRs:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **TPSR ID** | **Vendor** | **Component** | **Release** | **i18n Enabled/**  **Translation Ready** |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

Third-Party Software Comments:

Use this section to add comments or elaborate on the grid above, if necessary.

Click here to begin typing

#### Product and Build Integration

*Describe changes to the overall structure of the source control system and/or build order that result from new features, modifications to features and addition of common components or third party software. In addition, this section will guide the planning and sequence of product component development and integration based on dependencies, i.e.:*

* *Core components that should be developed and/or developed first*
* *Components that need to be developed in parallel*
* *Components that can be developed independently*

Click here to begin typing

### Interfaces

Identify consumers and suppliers of interfaces, public or private. The project team should be able to clearly articulate all known conflicts and alignment points in terms of the interfaces they use and/or plan to publish to reduce quality risks when integrating with other products.

The Stability Level should clearly denote the anticipated level of change for the interfaces that they publish, so the consuming teams can understand if the project intends to change the API they consume.

You should call out collections of interfaces, rather than each individual interface and only call out specific interfaces where required (e.g. the stability level is different)

Copy these tables for every collection of interfaces supplied and consumed by the product.

#### Interfaces Supplied by Project

|  |  |
| --- | --- |
| Collection of Interfaces (e.g. Client API) | |
| **Purpose:** (short description) |  |
| **API Dependencies:** |  |
| **Public or Private:** |  |
| **Stability Level:** (High, Medium, Low) |  |
| **New or Updated:** |  |
| **Interface Contract Required:** (Y/N) |  |

**Additional Comments:** Click here to begin typing

#### Interfaces Consumed by Project

|  |  |
| --- | --- |
| Collection of Interfaces (e.g. Client API) | |
| **Purpose:** (short description) |  |
| **Supplier Project Name and PRIME Clarity ID:** |  |
| **Supplier Product and Revision** |  |
| **Interface Contract Required:** (Y/N) |  |

**Additional Comments:** Click here to begin typing

### Certifications

Identify the design requirements associated with the Certification requirements for the project (e.g. FIPS 140-2, Common Criteria, SAP Certification). Compliance with a certification may require application changes or testing only. In the section below, define the design and implementation considerations. A list of certification requirements is included in the PRD.

#### <Certification n>

*Include design and implementation considerations for the certification.*

Click here to begin typing

### Other Dependencies

*Define any additional constraints that affect product design.* *, development, testing, environment and / or resources.*

Click here to begin typing

## Non-Functional Feature Design (AKA the abilities)

*Investment in the Non-Functional Requirements is a critical initiative for CA. For more information refer to the* [*CA Policy for Product Non-Functional Requirements*](http://qms.ca.com/document.asp?ID=9547) *(AKA “…Abilities”) in QMS.*

*Insert the* [*CA Abilities Scorecard*](http://qms.ca.com/documents/default.asp?P6=FIND&srchid=10216)*:*

**

Note: The score for *This Project* should reflect the level of compliance intended to ship

**Example**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *Ability* | *Granted by* | *Granted On* | *Reason for Exception* | *Plan for Remediation* |
|  |  |  |  |  |
|  |  |  |  |  |

**Note: Additional details on exceptions can be provided below along with details on compliance to the Ability.**

### ACCESSABILITY

All products developed by CA and common components must meet the global accessibility criteria, including Section 508 of the US Rehabilitation Act, and to ensure that a Voluntary Product Accessibility Template (VPAT) is created. Provide a high level description of how the product’s architecture and technologies will support the accessibility requirements defined in the PRD.

Click here to begin typing

### INTEGRATEABILITY

*All products developed by CA and common components should provide uniform mechanisms to integrate with internal and external products. Provide a high level description of how the product’s architecture and technologies will support the Integrate-ability requirements defined in the PRD.*

*The Integrateability design is covered in the “INTEGRATION, INTERFACES, AND DEPENDENCIES” section above.*

Click here to begin typing

### Interoperability

*All products developed by CA and common components should coexist in a graceful way with common components and agents that are likely to be on the same logical server in common customer environments. Provide a high level description of how the product’s architecture and technologies will support the Interoperability-ability requirements defined in the PRD.*

Click here to begin typing

### LocalizaBILITY

All products developed by CA and common components must comply with the [**Product Internationalization Policy**](http://qms.ca.com/document.asp?ID=9332). Provide a high level description of how the product’s architecture and technologies will support the internationalization requirements defined in the PRD

Click here to begin typing

### Scalability

*All products developed by CA and common components should be architected, designed and implemented to support customer load requirements as documented in the PRD. Provide a high level description of how the product’s architecture and technologies will support the Scalability requirements defined in the PRD.*

Click here to begin typing

### Securability

*Securability ensures that all products developed by CA meet a defined level of security, minimizing the risk of exploit. All products developed by CA are expected to provide secure operations, implying proper user authentication, access controls for critical functions and resources, ensuring privacy and activity auditing. This section is to ensure the designer has given comprehensive thought to security in and among deployed systems.*

Click here to begin typing

### Supportability

*All CA should implement a known and consistent set of capabilities to enable customers to effectively consume our products throughout the deployment lifecycle. Provide a high level description of how the product’s architecture and technologies will support the Supportability requirements defined in the PRD.*

Click here to begin typing

### upgradeability

All products developed by CA and common components should ensure that major releases can be upgraded in a controlled and secure way and without a large investment of effort by the customer. Provide a high level description of how the product’s architecture and technologies will support the Upgradeability requirements defined in the PRD.

*Indicate if any new licenses or new component codes are needed for upgrade.*

Click here to begin typing

### USABILITY

*All products developed by CA must be compliant with the CA User Interface Design Standards appropriate to the platform. Provide a high level description of how the product’s architecture and technologies will support the Usability requirements defined in the PRD.*

Click here to begin typing

### Supported Platforms

Define the design considerations for all Supported Platforms.

### A list of all platform versions/levels that the product/component will operate on is included in the PRD.

## PRODUCT design and Specifications

### Installation/Uninstall

Define the design and implementation considerations for product installation and uninstall. This section may include considerations for the install medium (CD, DVD, download), interfaces, installation sequence, coexistence on the same machine or LAN, etc.

Click here to begin typing

### Availability

Describe the Availability design considerations at the highest level.

This section may include design considerations for algorithm efficiency, memory usage, size of database inputs, network communications, database queries, crash resistance, handling of null values, etc.

Click here to begin typing

### Migration

*Define the design requirements for migration of existing products/components. Indicate if any new licenses or new component codes are needed.*

Click here to begin typing

### Performance

Describe the approach at the highest level. Elaborate on the technical direction to achieve performance goals. This may include algorithm efficiencies, automation, process changes, etc.

Click here to begin typing

### Product Licensing Schema

Define how the product will be licensed. Identify the impact of licensing changes.

## Design limitations, assumptions, and issues

This section should list current limitations and assumptions made in the design. These may include unique characteristics or testing requirements.

Click here to begin typing

### Limitation

Not support Linux sessions based on this design. Instant VM for Linux will be in separate document.

Instant VM need to inject the disk driver and network tools into the virtual disk. Since the proxy server is a Windows server it could not recognize the Linux file system. This is a challenge for Instant VM support Linux.

### Issues

Since the DDS is an evolving representation of the design, this section is used to keep track of issues and items that need special attention.

\* If PRIME Clarity is used to manage the issues or risks, include the Issue Name and ID, and the remaining fields can be blank.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Issue Name** | **ID\*** | **Description** | **Priority** | **Resolution** |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

# Internals

## Programming details

List procedures, modules and programs to be written, changed or deleted to meet the module specifications. Create as many subsections as needed for each module/program that is affected.

Click here to begin typing

### Module 1

Provide a description of the module, including the purpose, functionality, and interdependency. Describe major ideas and concepts that pertain to the design of this module. If this is an add-on feature, describe the design changes required in existing support modules.

Click here to begin typing

#### Detailed Implementation Description

Describe the implementation of the module in detail. Use diagrams as much as possible.

Click here to begin typing

#### External Interfaces to the Other Modules

Describe the external interfaces provided to other modules. Describe exported APIs and structure definitions, class definitions, etc.

Click here to begin typing

#### Unit Testing Description

List external functions and procedures that will be unit tested.

Click here to begin typing

### Module 2

Provide a description of the module including the purpose, functionality, and interdependency. Describe major ideas and concepts that pertain to the design of this module. If this is an add-on feature, describe the design changes required in existing support modules.

Click here to begin typing

#### Detailed Implementation Description

Describe the implementation of the module in detail. Use diagrams as much as possible.

Click here to begin typing

#### External Interfaces to Other Modules

Describe the external interfaces provided to other modules. Describe exported APIs and structure definitions, class definitions, etc.

Click here to begin typing

#### Unit Testing Description

List external functions and procedures that will be unit tested.

Click here to begin typing

## Internationalization

All CA products and common components must comply with the [CA Product Internationalization Procedure](http://qms.ca.com/documents/default.asp?P6=FIND&srchid=9332). Provide a description of how the component or design entity supports the internationalization requirements defined in the PRD. Additional information about these requirements can be found in the [Agile@CA Globalization Practice](http://cawiki.ca.com/display/AgileCA/Agile@CA+Globalization+Practice#AgileCAGlobalizationPractice-3224 ProductLocalization) and at the [Globalization Center](http://globalization.ca.com/).

Click here to begin typing

## Localization

Please refer to the [Agile@CA Globalization Practice](http://cawiki.ca.com/display/AgileCA/Agile@CA+Globalization+Practice#AgileCAGlobalizationPractice-3224 ProductLocalization) for information about Localization. In the section below, list all languages that you want your product localized into.

Click here to begin typing

## Updated effort estimate

The updated estimate derived from the DDS should confirm the estimates provided at Phase 1: Business Case Review. If the estimate is different, the Development Manager and Project Manager must be notified to determine the impact on the project schedule.

Click here to begin typing

# Impact Summary

The intention of this section is to give other groups such as Tech Info., Technical Support, Education, CA Services, QA, and Localization an idea of the impact of this development effort.

## Product/Component summary

List the menus, screens/panels, commands, reports, and messages that are impacted by the development of the module/function. Summarize these changes.

Click here to begin typing

## Documentation summary

List the existing end-user documentation that is affected by module changes; this includes structured information entities for Component Projects.

| **MANUAL** | **IMPACT** |
| --- | --- |
| Installation Guide |  |
| Administration Guide |  |
| Help Modules |  |
|  |  |

## Patent information

List any technology being developed for this feature that could be considered for a patent, and complete the Invention Disclosure Form. Information on the CA Innovation Patent Program and applicable forms can be found at [CA Patent Program](http://patent.ca.com/).

Click here to begin typing

# Quality Issues

Look at the component from the QA point of view. Suggest any special tests that will stress the component, think about how to make the component NOT work, and determine special tests that should be performed on this component. This is a guideline to the QA testing procedures.

## Testing recommendations and risk assessment

Suggest additional necessary function tests. Special test requirements include the security levels, hardware or software configurations, code page and multiple code pages, multi-system issues. Note anything that cannot be tested which might require field tests. What can go wrong? For example: Files are not allocated, TCP/IP bouncing, network errors. How are these situations handled? What are implications of failure in a component? Regression Risk assessment, migration and backward compatibility need to be considered.

Click here to begin typing

# Packaging and Installation Impact

This section should be used when the feature has an impact on the packaging or installation. If so, indicate and detail any special packaging or installation requirements. Include conversion utilities, compatibility, and migration issues for existing customers. Detail any new files that are required, as well as any new licensing requirements.

* Dependencies
* Configuration
* Install Paths
* Multi-language information
* Upgrade Information
* Registry Information
* Default Installation
* Licensing Information (also see [Licensing Requirements Checklist](http://qms.ca.com/documents/default.asp?srchID=1158))
* Common Component levels to be upgraded

Click here to begin typing

# technical Support Impact

Detail any diagnostics or trace facilities built in to the component, or facilities built into other components that would be used to gather information about what is happening. Note anything that may make the component:

* Difficult to diagnose (for example: no tracing facility)
* Difficult to service
* Unreliable (for example: External Risks)
* Workarounds
* Difficult to support (new skill set required)

Click here to begin typing