Public Works and Government Services Canada

Travaux publics et Services gouvernementaux Canada

RETURN BIDS TO:
RETOURNER LES SOUMISSIONS À:

Bid Receiving - PWGSC / Réception des soumissions - TPSGC
11 Laurier St. / 11, rue Laurier
Place du Portage, Phase III
Core 0A1 / Noyau 0A1
Gatineau
Québec
K1A 0S5
Bid Fax: (819) 997-9776

REQUEST FOR PROPOSAL
DEMANDE DE PROPOSITION

Proposal To: Public Works and Government Services Canada

We hereby offer to sell to Her Majesty the Queen in right of Canada, in accordance with the terms and conditions set out herein, referred to herein or attached hereto, the goods, services, and construction listed herein and on any attached sheets at the price(s) set out therefor.

Proposition aux: Travaux Publics et Services Gouvernementaux Canada

Nous offrons par la présente de vendre à Sa Majesté la Reine du chef du Canada, aux conditions énoncées ou incluses par référence dans la présente et aux annexes ci-jointes, les biens, services et construction énumérés ici sur toute feuille ci-annexée, au(x) prix indiqué(s).

Comments - Commentaires

Title - Sujet
SIMULATION ENGINE/GUI DEVELOPMENT

Solicitation No. - N° de l'invitation	**Date**
W7714-125376/A	2013-05-14

Client Reference No. - N° de référence du client
W7714-125376

GETS Reference No. - N° de référence de SEAG
PW-$$ZM-613-26040

File No. - N° de dossier	**CCC No./N° CCC - FMS No./N° VME**
613zm.W7714-125376	

Solicitation Closes - L'invitation prend fin
at - à 02:00 PM
on - le 2013-06-14

Time Zone
Fuseau horaire
Eastern Daylight Saving Time EDT

F.O.B. - F.A.B.
Plant-Usine: ☐ **Destination:** ☐ **Other-Autre:** ☐

Address Enquiries to: - Adresser toutes questions à:	**Buyer Id - Id de l'acheteur**
Pierre, Anoule	613zm

Telephone No. - N° de téléphone	**FAX No. - N° de FAX**
(819) 956-2137 ( )	(819) 956-1207

Destination - of Goods, Services, and Construction:
Destination - des biens, services et construction:
DEPARTMENT OF NATIONAL DEFENCE
NDHQ - DRDC - CORA
MAJOR GENERAL GEORGE R. PEARKES BLDG
101 COLONEL BY DRIVE, 8 FLR CBC
OTTAWA, ON
K1A0K2

Instructions: See Herein

Instructions: Voir aux présentes

Vendor/Firm Name and Address
Raison sociale et adresse du fournisseur/de l'entrepreneur

Delivery Required - Livraison exigée	**Delivery Offered - Livraison proposée**
See Herein	

Vendor/Firm Name and Address
Raison sociale et adresse du fournisseur/de l'entrepreneur

Telephone No. - N° de téléphone
Facsimile No. - N° de télécopieur

Name and title of person authorized to sign on behalf of Vendor/Firm (type or print)
Nom et titre de la personne autorisée à signer au nom du fournisseur/ de l'entrepreneur (taper ou écrire en caractères d'imprimerie)

Signature **Date**

Issuing Office - Bureau de distribution
Informatics Professional Services Division / Division des services professionnels en informatique
11 Laurier St., / 11, rue Laurier
3C2, Place du Portage
Gatineau
Québec
K1A 0S5

Canada

ANNEX A - STATEMENT OF WORK

RIGHT PERSON, RIGHT QUALIFICATIONS, RIGHT PLACE, RIGHT TIME, HUMAN RESOURCES TECHNOLOGY DEMONSTRATION PROJECT (R4 HR TDP)

DEVELOPMENT OF R4 HR APPLICATION

1.0 BACKGROUND

1.1 Defence Research and Development Canada (DRDC) existing modelling and simulation capability utilizes a wide array of Commercial-off-the-Shelf (COTS) and custom software tools developed both internally and by other allied nations who have made these tools available to Department of National Defence / Canadian Forces DND / CF through various collaborative arrangements. As a consequence, many of these software tools were originally developed to meet requirements that are not specific to DND / CF. For example, the Arena® simulation environment, which is widely used as a general simulation software application by DRDC, was originally developed by Rockwell's software division for the simulation of operations and logistics in a manufacturing and / or industrial setting. While these software tools do provide a powerful capability to facilitate the provision of timely and accurate analyses to DND / CF decision makers, they often exhibit limitations in addressing the unique challenges of managing DND / CF resources and in providing comprehensive analyses of DND / CF business processes.

1.2 These fundamental limitations in the capabilities of existing modelling and simulation tools include, but are not necessarily limited to, the following deficiencies:

a) Flexibility: The simulation requirements and scenarios required by DND / CF decision makers change frequently and demand rapid response, yet the complex models of DND / CF processes embedded in existing software tools are often difficult or impossible to change and update in a timely fashion;

b) Scalability: The scale of DND / CF resource simulations is much larger than those required by most commercial enterprises. As a consequence, most commercial software tools are not usually capable of handling the volume of data that must be processed in a typical simulation of standard DND / CF processes; and

c) Integration: DRDC maintains a wide range of highly developed models built on years of input from Operational Research (OR) analysts and military Subject Matter Experts (SME). However, each of these models is typically focused on a specific problem domain which cannot be readily integrated with models of other DND / CF processes to take into account the interactions between the different domains (e.g. the impact of increased deployments on the availability of instructors for training schools).

1.3 The R4 HR TDP team has developed a prototype of the proposed simulation environment (hereinafter, R4 HR prototype). The prototype includes an implementation of the simulation engine and Graphical User Interface (GUI). It was developed so that the R4 HR modelling and simulation concepts and user interface features could be tested. The prototype is functionally a 75% solution. It is possible to build and run simulations in the prototype. However, certain features were either not implemented or implemented in a "quick and dirty" fashion because of personnel resources limitations. The development of the prototype was focused on rapidly testing the functionality and paid minimal attention to proper testing or optimizing performance. As a result, the prototype is missing some interface features, the code is not well designed or well structured in many areas and contains some known problems (e.g. memory leaks), and the execution speed can be poor.

2.0 SCOPE

2.1 OBJECTIVES

The primary objective of the R4 HR TDP is to develop and demonstrate a simulation software environment (hereinafter, R4 HR application) that addresses the deficiencies identified in section 1.2. This must involve analysing the R4 HR prototype and redesigning and redeveloping the

prototype using professional software development practices to deliver a complete, high-quality simulation software application.

The two main components that the Contractor is to develop and deliver include:

a) A simulation engine that implements the scripting language programmed events, relational database technology and actor-oriented paradigm; and

b) A 3D GUI that facilitates the creation, execution, integration and management of simulation models and that provides integrated analysis and reporting capabilities to present simulation results in predefined, custom and interactive formats.

3.0 REQUIREMENTS

3.1 Overview

3.2 The objective of this contract is to be achieved by integrating technologies and modeling approaches that are ideally suited to simulating DND / CF HR-related processes which include:

a) The use of a scripting language to program discrete event simulation events and to allow simulation logic to be dynamically updated during simulation execution;
b) The incorporation of a relational database in the simulation and the ability to execute Structure Query Language commands on the database during simulation execution; and
c) The implementation of an actor-oriented modelling approach which encapsulates discrete event simulation logic and data including database tables in a modular construct called an actor.

3.3 The R4 HR application must allow analysts to build complex process models, conduct simulations consisting of multiple integrated models, and permit dynamic exploration of the results. The exploitation of this technology must enable DRDC to respond to more challenging and pressing HR issues in addition to providing DND / CF decision makers with advice based on more comprehensive HR-related analyses.

3.4 The R4 HR application must provide all of the functionality of the R4 HR prototype plus the additional features and functions identified in the Statement of Work (SOW) and its Appendices.

3.5 The R4 HR application must be designed and implemented such that it achieves a high level of performance in simulation execution and GUI responsiveness and has a high level of visual polish, usability and feature richness in the GUI. When implemented on the Target Production Environment, the R4 HR application must meet the performance requirements identified in the SOW and its Appendices.

3.6 Incremental Approach

The initial phase of the contract must be the development of the system architecture. This initial phase must be followed by the development of the R4 HR application. Three builds must be used to develop this application. Each build must be followed by a demonstration to allow the user community to identify the deficiencies of the most recent version of the R4 HR application.

Build 1 must provide the functionality specified in Appendix 4 of this Annex.

Build 2 (as and when requested, initiated by Task Authorizations only) must address the deficiencies identified by the user community during Build 1 and must add the functionality of optimizing simulation scenarios and providing tools to analyse simulation results.

Build 3 (as and when requested, initiated by Task Authorizations only) must address the deficiencies identified by the user community during Build 2 and must add a selection of advanced techniques. The deficiencies identified by the user community during Build 3 must be addressed during the final phase of the contract (Final R4 HR Application (initiated by Task Authorizations only).

3.7 System Architecture

The contractor is to develop the system architecture. The development of the system architecture must be executed in two steps. The first step must define the system specification. The second step must describe the system architectural design.

3.8 Build 1

3.8.1 Refer to Appendix 3, Phase 3 – Development of Build 1 Version of R4 HR Application. This document identifies the Scope, Quality Control, Project Management, and Deliverables associated with the required work for Build 1.

3.9 Build 2 (As and When Requested, Initiated by Task Authorizations Only)

3.9.1 The Contractor must perform the work after receipt of a Task Authorization, to make the refinements required to address the deficiencies identified by the user community during the demonstration for Build 1. The refinements must take the form of improvements to functionality developed during Build 1, adjustments or feature additions, alterations to the look and feel of the GUI, the addition of new GUI displays and functions, and generally any new requirement that arises from the practical use of the latest version of the R4 HR application.

3.9.2 The contractor may be tasked to add an optimization capability that will consist of an internal optimization module and/or a third party optimization tool. The optimization capability will execute optimization routines on simulation scenarios built in the R4 HR application. The optimization software must be able to perform the following actions: 1) to adjust certain variable aspects of a simulation within specified bounds; 2) to queue individual simulation scenarios to be executed on available processors so that scenarios can be executed in parallel; and 3) to retrieve simulation scenario results so that the optimization routine can compute further adjustments to simulation variables and queue the execution of additional scenarios.

In addition, the contractor may be tasked to provide tools to facilitate the analysis and interpretation of simulation results. This may include: 1) interactive exploration and visualization of simulation output data; and 2) aggregation and statistical analysis of results from multiple simulation iterations carried out in the R4 HR application.

3.9.3 Build 2 includes a demonstration of the R4 HR application to the user community. The Contractor must support the demonstration. The results of the demonstration will influence the tasks for Build 3.

3.10 Build 3 (As and When Requested, Initiated by Task Authorizations Only)

3.10.1 The Contractor will be tasked to make the refinements required to address the deficiencies identified by the user community during the demonstration for Build 2. The refinements will take

the form of improvements to functionality developed during Builds 1 and 2, adjustments or feature additions, alterations to the look and feel of the GUI, the addition of new GUI displays and functions, and generally any new requirement that arises from the practical use of the latest version of the R4 HR application.

3.10.2 The contractor may be tasked to add various advanced functionalities. These may include: an automated model building feature (where simulation logic can be programmed to build or expand other simulation logic) and an integrated agent based simulation capability. Other advanced techniques may be considered for possible implementation.

3.10.3 Build 3 includes a demonstration of the R4 HR application to the user community. The Contractor must support the demonstration. The results of the demonstration will influence the tasks associated with the final refinements and enhancements to R4 HR application.

3.11 Final R4 HR Application (As and When Requested, Initiated by Task Authorizations Only)

The Contractor will be tasked to make the refinements required to address the deficiencies identified by the user community during the demonstration for Build 3. The refinements will take the form of improvements to functionality developed during Builds 1, 2 and 3, adjustments or feature additions, alterations to the look and feel of the GUI, the addition of new GUI displays and functions, and generally any new requirement that arises from the practical use of the latest version of the R4 HR application.

4.0

INITIAL PHASES

The Contractor must deliver the following phases:

4.1 **Phase 1**

Appendix 1, Phase 1 – Project/Configuration Management Services, identifies the scope of the work associated with project and configuration management.

4.2 **Phase 2**

Appendix 2, Phase 2 – Development of System Architecture, identifies the scope of work associated with the development of the system architecture and the production of the software plan.

4.3 **Phase 3**

Appendix 3, Phase 3 – Development of Build 1 Version of R4 HR Application, identifies the scope of work for Build 1.

5.0 TARGET PRODUCTION ENVIRONMENT - SPECIFICATIONS

5.1 Target 1: The R4 HR application must operate on a modern PC or laptop for scientific applications with the following specifications:

a) Windows 7 (64bit)

b) 2.5 GHz dual-core CPU
c) 4 GB of RAM
d) Dedicated video card with 1 GB of video memory
e) 500 GB of storage

5.2 Target 2: The R4 HR application must be optimized to operate on a high-performance computer with the following specifications:

a) Windows 7 (64bit)

b) 2 x 3GHz 6-core CPUs (12 total cores)
c) 48 GB of RAM
d) Dedicated video card with 1 GB of video memory
e) 500 GB of storage

5.3 Target 3: The R4 HR application must be able to execute on a very high-performance computer with the following specifications:

a) Windows HPC Server 2008

b) 8 compute nodes, each node having

- 2 x 3GHz 6-core CPUs (12 total cores)
- 48 GB of RAM
- 500 GB of storage

5.4 Acceptance testing will be carried out on Target 2, per Appendix 5 – R4 HR Application Performance Specifications. The contractor will also be required to demonstrate that the R4 HR application can operate on Targets 1 and 3 on DND-supplied equipment. The demonstration must be carried out in the presence of R4 HR TDP team.

5.5 For systems with multiple CPU cores, the R4 HR application will distribute various instances of the simulation to the available cores for parallel execution. Instances executed in parallel may be identical replications (for models with stochastic effects), different pre-defined scenarios, and/or automatically generated scenarios used in an optimization algorithm. The R4 HR application will also make use of multiple CPU cores as required to increase performance of the modelling GUI and output features.

6.0 QUALITY CONTROL

6.1 The Contractor must perform all the verification and validation necessary to substantiate that the software and/or material provided conforms to the requirements of any tasks that are pursuant to the Work. All components of the R4 HR application must be tested to confirm that the R4 HR application meets the functional and performance specifications described in the SOW and its Appendices. The Contractor must provide the following documents for all work involving the testing of software:

a) Software Test Plan (DID ENG-010 and CDRL 2010)
b) Software Test Description (DID ENG-011 and CDRL 2011)
c) Software Test Report (DID ENG-012 and CDRL 2012)

6.2 The Contractor must demonstrate that the delivered software replicates the functionalities of the prototype and produces identical output. The Contractor must also demonstrate that the delivered software meets the performance requirements identified in Appendix 5 – R4 HR Application Performance Specifications.

6.3 The R4 HR application will be tested by DRDC against the R4 HR prototype to confirm that the R4 HR application is functionally correct. The R4 HR TDP team has and will develop various test scenarios that will be used for this purpose. The supplied scenarios will include scenarios that are representative of real-world DND / CF HR simulations in scale and complexity. These scenarios must be used to profile execution speed and optimize the performance of the simulation engine. The R4 HR TDP team will witness the loading and running of simulation scenarios that have been pre-built in the prototype by the R4 HR TDP team and that will be used to test the R4 HR application. The exact behaviour and outputs of these test runs will be compared to identical runs performed in the prototype, and the requirement is that these results must be identical. The R4 HR application must be able to export the end state of a completed simulation back to the prototype scenario file format so that it can be opened again in the prototype.

7.0 PROJECT MANAGEMENT

7.1 The work must be initiated as detailed in Appendix 1, Phase 1 – Project/Configuration Management Services. Table 1 provides the proposed schedule for the development and testing of the R4 HR application. The Contractor is encouraged to suggest changes to the proposed schedule, if deemed more appropriate and optimal for the success of the Contract. The schedule is based on a Contract Award date of September 2nd, 2013. If this date changes, the schedule

must be amended accordingly. However, if the Contract Award date is later, all efforts must be made to still have a completion date of March 31st, 2016.

7.2

Table 1. Proposed Schedule

Events	**Start**	**End**
Configuration Management Plan	September 2013	October 2013
GO / NO GO Decision Point		
System Architecture	September 2013	November 2013
GO / NO GO Decision Point		
Build 1	December 2013	August 2014
GO / NO GO Decision Point		
Build 2	September 2014	April 2015
GO / NO GO Decision Point		
Build 3	May 2015	December 2015
GO / NO GO Decision Point		
Final R4 HR Application	January 2016	March 2016

7.3 A Progress Review Meeting (PRM) must be held on a monthly basis for the duration of the Contract. This meeting must involve the R4 HR TDP team and the appropriate personnel from the Contractor. The PRM aims at reviewing the effort made from a management perspective for a specific period of time. The frequency will be adjusted by the Technical Authority depending on the needs to track progress at different stages of the project. The PRM must be brief and focus on providing a status of recent activities and issues without addressing detailed technical questions. The Contractor is responsible for preparing the agenda and publishing the minutes of the PRM. The agenda must be submitted to the Technical Authority at least two (2) working days prior to the meeting. The minutes must be submitted to the Technical Authority no later than five (5) working days after the meeting. The agenda and minutes must be endorsed by the Technical Authority.

7.4 The initial phase must include the following technical reviews: System Requirements Review (SRR) and System Design Review (SDR). Each build must include the following technical reviews: Preliminary Design Review (PDR), Critical Design Review (CDR) and Test Readiness Review (TRR). Each review must involve the R4 HR TDP team and the appropriate personnel from the Contractor. Each review could last up to one day and must address all technical issues related to the milestone. The Contractor is responsible for preparing the agenda and publishing the minutes of the reviews. The agenda must be submitted to the Technical Authority at least two (2) working days prior to the meeting. The minutes must be submitted to the Technical Authority no later than ten (10) working days after the meeting. The agenda and minutes must be endorsed by the Technical Authority. The PDR and CDR will be used as GO/NO GO decision points.

7.5 Ad hoc Reporting: The Contractor must report to the Technical Authority, within two (2) working days of its identification, any element that may have a significant impact on the progress of the

project. Such problems must be identified, verbally or in writing, to the Technical Authority when any of the following circumstances or situations arises: 1) A significant issue that may impact the objectives of the project; 2) A significant managerial issue that may impact cost, schedule or the technical quality of the deliverables; or 3) A schedule slippage for a milestone or a deliverable is anticipated. The problem must also be addressed at the next PRM or earlier if deemed necessary.

8.0 DELIVERABLES

8.1 Overview

Deliverables are identified in Article 6 of, in each of the following appendices:

- Appendix 1 (Project/Configuration Management Services);
- Appendix 2 (Development of System Architecture);
- Appendix 3 (Development of Build 1 Version of R4 HR Application);
- Or as Specified in each Task Authorization.

Appendix 7 (Data Deliverables) provides the Contract Data Requirements List (CDRL) and the Standard Data Item Description (DID) for the deliverables.

8.2 Developed Software and Documentation:

a) For all developed software, the Contractor must provide an installable version of the software that can be installed on any hardware matching the specifications of the target production environments, and the Contractor must provide all source project files including source code and any accompanying resource and configuration files. Instructions for building the software from the source files must also be included. DND will supply an external portable hard drive on which the contractor must store the deliverables. The Contractor must then return the hard drive with the files to the R4 HR Team.

   Deliverables must be saved in one of the following file formats:

   - Windows binaries for executables
   - Text files for source code
   - MicroSoft Word files for documents
   - JPEG, GIF, BMP or PNG files for images
   - SQLite database files for databases

   Other file formats required for deliverables may be used, but the Contractor must inform the Technical Authority and the Technical Authority must agree to their use in order for deliverables to be accepted in other formats.

b) The Contractor must also supply the Software User Manual (DID ENG-013 and CDRL 2013) and Computer Programming Manual (DID ENG-014 and CDRL 2014).

8.3 FINAL DELIVERABLES:

The Contractor must deliver a simulation environment comprising of:

a) A simulation engine that implements the scripting language programmed events, relational database technology and actor-oriented paradigm; and

b) A 3D GUI that facilitates the creation, execution, integration and management of simulation models and that provides integrated analysis and reporting capabilities to present simulation results in predefined, custom and interactive formats.

9.0 GOVERNMENT FURNISHED EQUIPMENT (GFE)

DRDC will supply the Contractor with a copy of the R4 HR prototype and pre-built simulation scenarios. The Contractor is to build the R4 HR application using Contractor supplied hardware and software.

10.0 APPENDICES:

Appendix 1 – Phase 1 – Project and Configuration Management Services

Appendix 2 – Phase 2 – Development of System Architecture

Appendix 3 – Phase 3 – Development of Build 1 Version of R4 HR Application

Appendix 4 – R4 HR Application Functional Specifications

Appendix 5 – R4 HR Application Performance Specifications

Appendix 6 – Lexicon/Abbreviations

Appendix 7 – Data Deliverables

APPENDIX 1 TO ANNEX A

PHASE 1 – PROJECT/CONFIGURATION MANAGEMENT SERVICES

1.0 SCOPE

The purpose of Phase 1 is to provide project and configuration management services for the period of the contract. Phase 1 must commence immediately following contract award and last for the duration of the contract.

2.0 APPLICABLE DOCUMENTS

No document is referenced in Phase 1.

3.0 SCOPE OF WORK

3.1 Implementation

3.1.1 The Contractor must provide the details of the Implementation Management Plan and manage the work during the life of the contract based on this plan.

3.1.2 The Contractor must provide the details of the risk management plan and manage the risks during the life of the contract based on this plan.

3.1.3 The Contractor must provide the details of the quality management plan and manage the quality of the work and deliverables during the life of the contract based on this plan.

3.1.4 The Contractor must provide the details of the communication plan and manage communications during the life of the contract based on this plan.

3.2 Meetings and Reviews

3.2.1 A contract kick-off meeting must be held within two weeks of contract award to discuss in detail contract requirements and deliverables. The kick-off meeting must be organized by the Contractor at a location acceptable to the Technical Authority.

3.2.2 The Contractor must conduct progress review meetings in accordance with section 7.3 of the SOW.

3.2.3 The Contractor must conduct technical reviews in accordance with section 7.4 of the SOW.

3.2.4 The Contractor must issue ad hoc reports in accordance with section 7.5 of the SOW.

3.3 Project Planning

The Contractor must plan the work to be conducted, develop the work breakdown structure (WBS) and provide a master schedule of the activities.

3.4 **Project Budgeting**

The Contractor must keep track of the financial resources allocated regarding all work approved under this contract.

3.5 **Configuration Management**

3.5.1 The Contractor must produce the Configuration Management Plan and must exercise configuration control during the life of the contract in accordance with this plan.

3.5.2 The contractor must perform audits of configuration in terms of content and readiness for release.

3.5.3 The contractor must report the status of the R4 HR application baseline and its component configuration items.

3.5.4. The contractor must provide version controls of all work products ensuring versions are stored and retrievable.

4.0 QUALITY CONTROL

The quality control must be conducted as per section 6.0 of the SOW.

5.0 PROJECT MANAGEMENT

The project management must be conducted as per section 7.0 of the SOW.

6.0 DELIVERABLES

6.1 The deliverables for Phase 1 are as follows:

6.1.1 Project Management: The Contractor must provide an Implementation Management Plan (DID PM-001 and CDRL 1001) no later than one (1) month after Contract Award.

6.1.2 Meeting Agenda: The Contractor must prepare a meeting agenda for progress review meetings and technical reviews in accordance with section 7.0 of the SOW. The progress review meetings must cover the following as a minimum:

a) Payment status, contract value and contract invoicing status;
b) Current status of all technical and logistical problems, and actions taken to resolve them;
c) All events/milestones scheduled, achieved or missed must be covered and reasons provided for missed milestones;
d) Target completion date for each WBS;
e) A summary of future plans, recommendations and proposals both for the next reporting period and for any long term plans; and
f) The status of action items, noting which action items have been completed, are on-going, are superseded or replaced, as appropriate. The action item list must show the person or position responsible for completing the action, due date and date completed/closed.

6.1.3 Meeting Minutes: The Contractor must prepare meeting minutes for progress review meetings and technical reviews in accordance with section 7.3 and 7.4 of the SOW.

6.1.4 Project Schedule: The contractor must prepare a Project Schedule that shows all project activities from contract award through to final acceptance and hand over. The Project Schedule is updated regularly to provide the Technical Authority with visibility of project accomplishment to date at a

summary level which is indicative of overall project performance. The Project Schedule provides an overview of the entire project scope and its status at given points in time. For all work to be performed, the contractor must provide a clear and concise definition of the work to be accomplished, and by whom it is to be performed. The Project Schedule must be delivered to the Technical Authority as part of the Implementation Management Plan and updated on a monthly basis. The Technical Authority will review and accept or reject the Project Schedule within twenty (20) working days from its reception. The Master Project Schedule must be produced in a format acceptable to the Technical Authority. The standard software tool for project scheduling in DND is Microsoft© Project.

6.1.5 Configuration Management: The Contractor must provide a Configuration Management Plan (DID ENG-001 and CDRL 2001) no later than one (1) month after Contract Award.

6.1.6 Configuration Identification Report: The contractor must prepare a Configuration Identification Report to document the identity of the R4 HR application baseline and configuration items that must be controlled. The Configuration Identification Report must be delivered to the Technical Authority one (1) month after the start of Build 1. It must be provided as an electronic copy (in Microsoft® Word format) with three (3) printed copies. The Technical Authority will review and accept or reject the Configuration Identification Report within twenty (20) working days from its reception.

6.1.7 Baseline Report: The contractor must prepare and update on a monthly basis a Baseline Report to document the initial and updated status (e.g. version) of each configured item identified as an element to be controlled as part of the R4 HR application baseline, the release readiness of each controlled element, and an electronic log of the versions of software component items (e.g. source files) under automatic version control. The first version of the Baseline Report must be delivered to the Technical Authority two (2) months after the start of Build 1. The initial version and each subsequent version of the Baseline Report must be provided as an electronic copy (in Microsoft® Word format), with three (3) printed copies. The Technical Authority will review and accept or reject the Baseline Report within twenty (20) working days from its reception.

APPENDIX 2 TO ANNEX A

PHASE 2 - DEVELOPMENT OF SYSTEM ARCHITECTURE

1.0 SCOPE

The purpose of Phase 2 is to develop the system architecture and to produce the software development plan. Phase 2 must commence immediately following contract award.

2.0 APPLICABLE DOCUMENTS

2.1 The R4 HR TDP team will provide the contractor with applicable reports and presentations that describe the motivation, objectives and applications areas of the R4 HR application.

2.2 The R4 HR TDP team will provide the contractor with the R4 HR prototype in the form of Python 2.7 source code, resource files, example simulation scenarios, installation instructions and usage instructions.

3.0 SCOPE OF WORK

3.1 The Contractor must perform the necessary work to develop the system/subsystem specification.

3.2 The Contractor must perform the necessary tasks to develop the system/subsystem design description.

3.3 The Contractor must produce the software development plan.

4.0 QUALITY CONTROL

The quality control must be conducted as per section 6.0 of the SOW.

5.0 PROJECT MANAGEMENT

The project management must be conducted as per section 7.0 of the SOW.

6.0 DELIVERABLES

6.1 The deliverables for Phase 2 are as follows:

6.1.2 Technical Documentation: The Contractor must provide the following documents:

a) System/Subsystem Specification (DID-ENG-003 and CDRL 2003) at least ten (10) working days prior to the System Requirements Review;
b) System/Subsystem Design Description (DID-ENG-004 and CDRL 2004) at least ten (10) working days prior to the System Design Review; and
c) Software Development Plan (DID ENG-002 and CDRL 2002) at least ten (10) working days prior to the System Design Review.

APPENDIX 3 TO ANNEX A

PHASE 3 - DEVELOPMENT OF BUILD 1 VERSION OF R4 HR APPLICATION

1.0 SCOPE

The purpose of Phase 3 is to develop and demonstrate Build 1 version of the R4 HR application. Phase 3 must commence upon successful completion of System Design Review.

2.0 APPLICABLE DOCUMENTS

2.1 The R4 HR TDP team will provide the contractor with various simulation scenarios to be used to measure the performance of the R4 HR application relative to the performance requirements identified in Appendix 5 – R4 HR Application Performance Specifications.

3.0 SCOPE OF WORK

3.1 The Contractor is to develop the simulation engine component of the R4 HR application. In other words, given a pre-existing simulation scenario, the simulation engine delivered by the Contractor must execute this scenario, running the simulation until it completes. The simulation engine must implement the scripting language, relational database and actor-oriented design specified in Appendix 4 and work within the “Target Production Environment” identified in this SOW.

3.2 The simulation engine component must include all the simulation component functionality and features identified in Appendix 4 – R4 HR Application Functional Specifications and the performance requirements identified in Appendix 5 – R4 HR Application Performance Specifications.

3.3 The R4 HR TDP team must witness the testing of the simulation engine using measures that indicate its correctness in executing scenarios, its support for required features, and its speed of execution.

3.4 As part of the Acceptance Testing, the Contractor must demonstrate to the R4 HR TDP team that the delivered simulation engine carries out an identical sequence of events and produces identical simulation output to the R4 HR prototype in executing the pre-built scenarios. The Contractor must also demonstrate to the R4 HR TDP team that the delivered simulation engine achieves in all these scenarios the performance specified in Appendix 5 – R4 HR Application Performance Specifications.

3.5 The Contractor is to develop the 3D GUI component of the R4 HR application for the building, execution, integration and management of models and the reporting of simulation output based on the specifications in Appendix 4 and working within the “Target Production Environment” identified in this SOW.

3.6 The 3D GUI must include all the functionality and features identified in Appendix 4 – R4 HR Application Functional Specifications and the performance requirements identified in Appendix 5 – R4 HR Application Performance Specifications.

3.7 The 3D GUI must provide a 3D visual representation of the model as it is developed and as it executes. It must allow the user to view and interact with simulation objects in 3 dimensions. It must provide interfaces to define and manipulate all aspects of the simulation, and to display graphical output.

3.8 The delivered R4 HR application must match or exceed all capabilities and features of the R4 HR prototype. In instances where Appendix 4 – R4 HR Application Functional Specifications describes capabilities and/or features that differ or are missing from the R4 HR prototype, Appendix 4 shall take precedence.

3.9 The R4 HR TDP team must participate in the evaluation of the 3D GUI using measures that indicate its ease of use, its support for required features, and its level of visual polish.

3.10 Build 1 includes a demonstration of the R4 HR application to the user community. The Contractor must support the demonstration. The results of the demonstration will influence the tasks for Build 2.

4.0 QUALITY CONTROL

The quality control must be conducted as per section 6.0 of the SOW.

5.0 PROJECT MANAGEMENT

The project management must be conducted as per section 7.0 of the SOW.

6.0 DELIVERABLES

6.1 The deliverables for Phase 3 are as follows:

6.1.2 Technical Documentation: The Contractor must provide the following documents:

a) Software Requirements Specification (DID ENG-005 and CDRL 2005) at least ten (10) working days prior to the Preliminary Design Review;
b) Interface Requirements Specification (DID ENG-006 and CDRL 2006) at least ten (10) working days prior to the Preliminary Design Review;
c) Software Design Description (DID ENG-007 and CDRL 2007) at least ten (10) working days prior to the Critical Design Review;
d) Interface Design Description (DID ENG-008 and CDRL 2008) at least ten (10) working days prior to the Critical Design Review;
e) Database Design Description (DID ENG-009 and CDRL 2009) at least ten (10) working days prior to the Critical Design Review;
f) Software Test Plan (DID ENG-010 and CDRL 2010) at least ten (10) working days prior to the Test Readiness Review;
g) Software Test Description (DID ENG-011 and CDRL 2011) at least ten (10) working days prior to the Test Readiness Review; and
h) Software Test Report (DID ENG-012 and CDRL 2012) no later than ten (10) working days after the conduct of the tests.

6.1.3 Developed Software and Documentation:

a) The Contractor must provide an executable copy of the software and a copy of all source code at the end of Phase 3.
b) The Contractor must also supply the Software User Manual (DID ENG-013 and CDRL 2013) and Computer Programming Manual (DID ENG-014 and CDRL 2014) at the end of Phase 3.

APPENDIX 4 TO ANNEX A

R4 HR APPLICATION FUNCTIONAL SPECIFICATIONS (Build 1 Only)

1.0 INTRODUCTION

This Appendix describes the required functionality of the R4 HR application. The R4 HR prototype that will be provided to the contractor implements most of the functionality identified in this Appendix and will serve as a reference implementation that the contractor can refer to for a detailed understanding of the intended functionality.

MANDATORY SPECIFICATIONS:

The R4 HR application must provide all of the features and functions identified in this Appendix.

The Appendix is broadly organized by component: simulation engine and 3D GUI. The simulation engine executes discrete event simulation scenarios. The 3D GUI allows the user to build discrete event models of a given system, to visualize the events occurring during a simulation run, and to view and analyse simulation output.

The functional specifications for Build 2 and Build 3 are not presented here because the details of those Builds will be defined during Build 1. The Task Authorization process will be used ("as and when requested") to present the functional specifications for Build 2, Build 3 and the final R4 HR application.

2.0 SIMULATION ENGINE

The simulation engine executes discrete event simulations. This means that the simulation consists of a finite number of events that occur at specific times. In a human resources context, examples of events are recruitment, promotion, starting a training activity, completing a training activity, being posted to a new position, and retirement. In general, events consist of a sequence of instructions that can modify the state of various aspects of the simulation and schedule other events to occur in the future. Once an event completes, the simulation engine selects the next pending event to process and advances the simulation clock to the time of that event.

The R4 HR application allows the user to create discrete event models by placing basic building blocks called "parts" and connecting them together using "wires" which are directional links from one part to another. Creating a wire from one part to another allows the former to access and modify the latter. The following section describes parts in detail and describes the effect of wiring parts together.

2.1 Parts

Parts are the active elements used to build a model. Each part technically consists of a standard container object (used by all parts) and an active object that is held by the container. The container allows the active object it holds to be a part of the R4 HR simulation environment. The active object can be one of many available types that determine the part type.

All parts appear in 3D space in the R4 HR GUI. They display their contents (which is the active object) surrounded by a frame with a symbol in the upper left corner that represents the part type, a name (that needs not be unique), optional shortcut buttons in the upper right corner, and wires connecting to and from the part. The aliases given to wires connecting from one part to other parts must be unique. Parts are hierarchically organized and keep track of their parent and their children. The standard part container stores the attributes common to all parts: the part's name, 3D position, parent part, child parts, connecting wires, and other visual attributes. The part's

active object stores any information unique to that part type. For example, the active object in a clock part is a clock object that contains the clock's time in ticks and the calendar time (year, month, day, hour, minute and second).

The name of a part is purely descriptive and has no bearing on the simulation logic. Therefore, different parts can share the same name. Within the model logic, a part is not referred to by its name but by an alias given to it by another part via a connecting wire. Thus each part provides an alias for each other part that it connects to, and a single part may be linked to by multiple other parts each using a different alias. This scheme allows model logic to be portable and easily integrated with other model logic because parts do not need to be protected against naming-collisions, and any part can link to any other part, regardless of its name, using its own alias.

There are currently 15 part types, however this number may increase in future builds of R4 HR application. The type is determined by the active object that the part contains. The rest of this section will describe the function of each of these parts.

Table 1. List of part types and symbols

Symbol	Part Type
	**Code**
	**Variable**
	**Data**
	**Sheet**
	**Table**
	**SQL**
	**Clock**
	**Hub**

	Actor
	Node
	Socket
	Button
	Multiplier
	Output
	Comment

With the exception of hubs, nodes, sockets, and multipliers that use custom graphics, all parts are displayed in a standard way consisting of the part contents surrounded by a frame, a title bar, the symbol for the part type in the upper left corner (from Table 1), a name immediately to the right of the symbol, and optional shortcut buttons in the upper right corner. Figure 1 illustrates the appearance of a typical part. This is how parts are displayed in the GUI, and they will be shown in this form throughout this section to illustrate part functionality.

Note: The illustrations used in this appendix do not necessary use a visual style consistent with Figure 1 above, as they have been taken from various past versions of the prototype.

For Build 1, the R4 HR application must reproduce the appearance of the current R4 HR prototype provided to the contractor at the time of contract award unless the contractor proposes and the technical authority accepts an improved visual style for the application.

The following section will describe the function of each of the part types in detail.

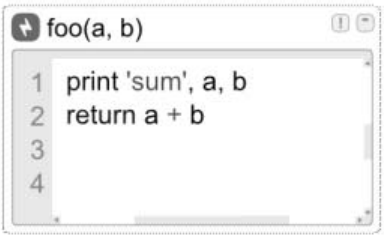

Figure 1. Standard part display showing part contents (in this case, Python 2.7 code with line numbers) and a surrounding frame with title bar consisting of the part symbol, the part name, and shortcut buttons.

2.1.1 Code Part

The code part is the most fundamental part used to build models. It contains a Python 2.7 script that represents the instructions to be carried out in an event. The scripts that exist in code parts define all the events to be performed by the model. The scripts are, in fact, the body of a function. As such, the code part also has defined parameters that can be used to pass information to the script, and the script may return a value. The code part can be wired to any other part type such as a data part or a clock part. If a code part is wired to another part, the script then has access to that part using the alias specified by the wire. In other words, aliases of the wires that connect a code part to other parts constitute the namespace of the code part as shown in Figure 2.

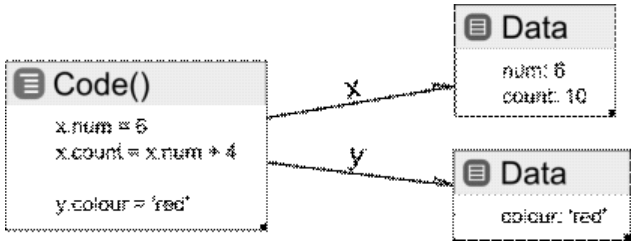

Figure 2. A code part accessing two data parts with aliases “x” and “y.” Note, this figure shows an older, outdated, visual style from the R4 HR prototype.

When one code part is linked to another code part, the former can call the latter as a function optionally passing in parameters and receiving the return value as shown in Figure 3.

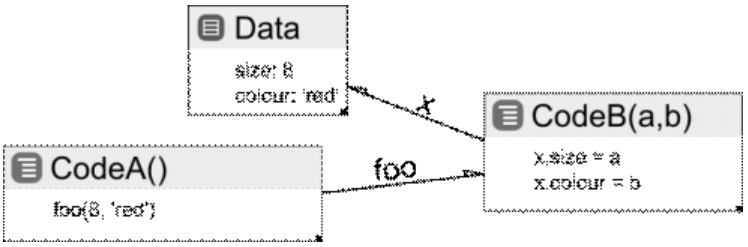

Figure 3. A code part (CodeA) calls a second code part (CodeB) as a function, passing parameters which are then stored in a data part.

A code part can also schedule another code part to execute at any time greater than or equal to the current time. This is achieved by calling a signal function which has the following definition:

signal(Code, Params=(), Priority=0, Time=None)

Amendment: The parameters of this function have been reordered as follows: Code, Params, Time, Priority.

where Code is the alias of the target code part to be scheduled for future execution, Params is a tuple consisting of the parameters to be passed to the target code part, Priority is the priority of the signal, and Time is when the target code part will execute. If the Time parameter is not specified, then it will default to the current simulation time. In order to specify the Time parameter, a clock part must be used. For example, if the code part is wired to a clock part with the alias

“clk,” the script can schedule another code part (with alias “foo”) to run at time 5 on the clock as shown in Figure 4.

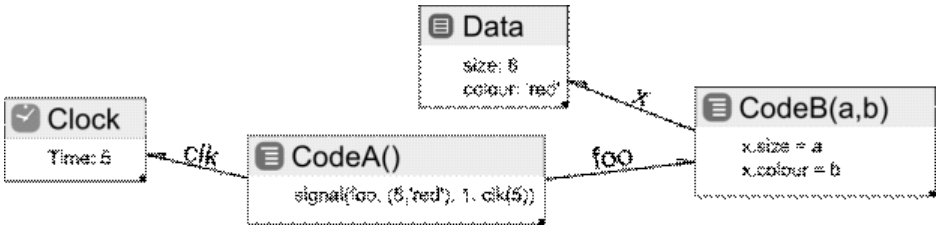

Figure 4. A code part (CodeA) schedules another code part (CodeB) to run at time 5 on the Clock.

The combination of a code part, its parameters, a priority and a time according to a clock constitutes a scheduled event. The signal function places this event on a queue of all pending events in the simulation. Because the scheduled code will not execute until a future time, the return value of the scheduled code is not returned to the code that called signal.

When a code part is created, the objects shown in Table 2 are automatically added to its namespace using the aliases shown (this list will grow in future versions). These are special internal objects to which every code part has access, and which do not appear in the GUI.

Table 2. List of objects that are automatically added to the code part namespace.

Alias	Object
Signal	The signal function described above
Math	The Python 2.7 math module
Random	The Python 2.7 random module
self	A reference to itself so that the code part can call itself or signal itself recursively.
admin	A container class instance that gives the code part access to various administrative functions and attributes of linked parts.
new_part	A function that accepts a part type, an alias and a position as parameters and creates a new part of that type with that alias at that position.
del_part	A function that accepts an alias as a parameter and deletes the part that is linked to with that alias.
set_sim_state	A function that sets the simulation to be in one of four execution states (described in detail in Section 2.2): ‘pause,’ ‘run-interactive,’ ‘run-realtime,’ or ‘run-noninteractive.’

log	A function to write custom text to the global log file.

The admin object in Table 2 allows a code part to access special attributes and methods of other parts that it links to. For example, if a code part is wired to a data part with alias “x,” the admin object allows the code part to change certain attributes of the data part including its name, position, scale, and display properties such as whether it is hidden or bolded. In this example, the syntax for changing the location of the data part along the x-axis would be “admin.x.PosX = 10.0.”

In most cases, the user will not need to access the administrative features of a part from code because these features can all be manipulated through the GUI (e.g. to change the position of a part, the user can drag it in the GUI). However, for advanced models, the user may want to animate the events occurring during the simulation, or the user may want to automate the creation of a model. In these cases, the admin object will be necessary.

The new_part function in Table 2 can be used to create a new part from code. The definition of the new_part function is:

new_part(part_type, alias, position)

The part_type parameter is a string that may be any of the names in Table 1. The alias parameter is a string that defines the name of the new part. The position parameter is a three-tuple, e.g. (5.0, 2.5, 8.1), specifying where the new part will be created in 3D space. After creating the part, the new_part function also creates a wire from the current code part to the new part giving it the specified alias. This adds the new part to the code part’s namespace so that the code can access the new part. 5 shows an example of one code part creating another code part using the new_part and admin functions.

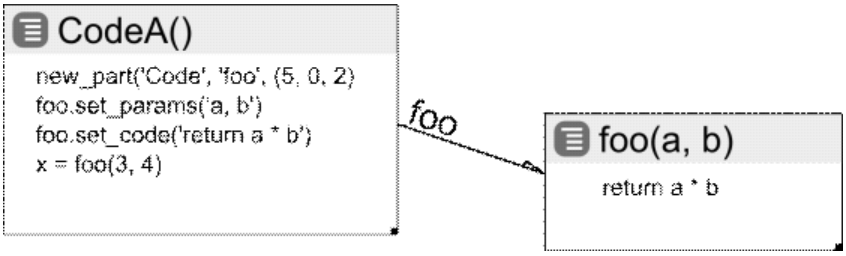

Figure 5. A code part creates a new code part (foo) using the new_part function and then uses the admin function to set the parameters and the code script of the new code part.

A code part can also create a link between two other parts that it is already wired to. All parts that can be wired to other parts will include the administrative method “link_part” with the following definition:

link_part(alias, part)

The alias is a string parameter specifying the alias of the wire to be created and, the part parameter is the part that will be linked to. Figure 6 shows an example of using a code part to create a wired connection between two other parts.

A code part can also delete a link between two other parts by calling the administrative function “unlink_part” passing the wire alias as a parameter. For example, in Figure 6, the code part on the left could delete the link “f” by executing the following:

admin.bar.unlink_part(‘f’)

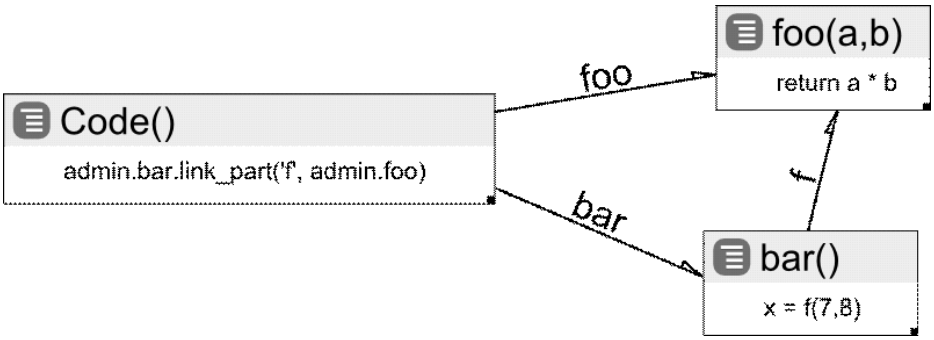

Figure 6. A code part creates a link from one code part (bar) to another code part (foo) by calling the administrative function “link_part.”

A code part can delete another part that it is linked to by calling the “del_part” function shown in Table 2. The “del_part” function has the following definition:

del_part(alias).

The alias parameter is the alias of the wire connecting to the part to be deleted.

2.1.2 Variable Part

The variable part is a catch-all part that is used to contain any Python 2.7 object that is not one of the other built-in types described in this section, such as a number, a string, a list, a dictionary, or a custom class. Code parts that link to a variable part can then interact with the variable part. For example, if the variable part contains a list as shown in Figure 7, a code part that links to it with alias “x” can append to the list by calling x.append().

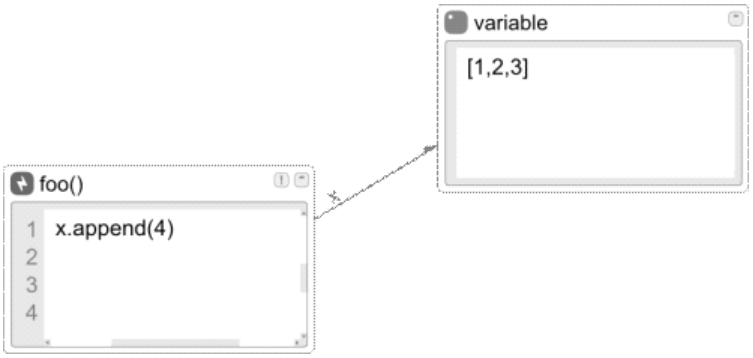

Figure 7. A variable part contains a Python 2.7 list, and the code part can interact with the list using the alias “x.”

The object contained in the variable part can be set by calling the admin function, “set().” For example, to change the list object in Figure 7 to a string, the code part would execute the following:

admin.x.set(‘Hello World!’)

2.1.3 Data Part

The data part is used to store variables. It consists of a simple container class. Data elements are member variables that are added to or removed from the container class. When a code part is wired to a data part it can access the container class using the wire alias as shown above in Figure 2. A code part can add new member variables to the data part by assigning them. For example, if a code part is wired to a data part with alias “vars,” a new variable name “size” can be created and assigned a value as follows:

vars.size = 20

Variables can be deleted from the data part using the Python 2.7 del function, for example:

del vars.size

2.1.4 Sheet part

The sheet part is similar to a data part except that it stores values in a two-dimensional grid of cells similar to a spreadsheet. Cells within the sheet can be indexed from a linking code part using 0-based array indexing, for example “sheet[0,2] = 5” would set the first row, third column cell to 5. Python 2.7 slice notation can be used to select ranges within the sheet. For example “sheet[:2, :2] = sheet[:2,8:10]” will copy the 2x2 range from one area of the sheet to another.

Sheet cells can also be indexed using MS Excel-style column-letter, row-number ranges. For example, the first-row, third-column would be indexed using “sheet[‘C1’].” Excel-style ranges, e.g. “sheet[‘A1:C8’] are also supported. Sheet columns can also be given optional custom names which may be used in place of the column letter. The “set_named_column()” function can be used set a column name. For example, if a sheet is given a custom column named “rank,” the fifth element within with this column can be indexed using “sheet[‘rank5’].”

Sheet parts also support importing from and exporting to MS Excel sheets.

2.1.5 Table Part

The table part contains a relational database table that exists in a database integrated with the R4 HR application. The table part also stores the names and data-types for the table fields. The table part provides functions to perform basic table operations such as inserting data, adding and removing fields, changing field names and selecting rows from the table that match specified criteria. When a code part is wired to a table, it can access the table using the alias of the wire and calling these functions as shown in Figure 8.

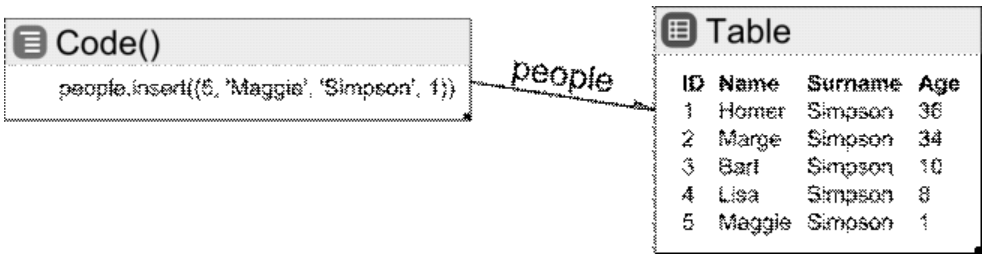

Figure 8. A code part that inserts a record into a table part.

Table parts also provide functionality to import from and export to external databases including MS Access or SQL Server.

2.1.6 SQL Part

The SQL part contains a structured query language (SQL) command. The SQL command acts on tables that the SQL part is wired to using the wire aliases. The SQL part also has parameters that can be used in the SQL command. For example, if the SQL part has a parameter "pSurname" and is wired to a table part with alias "people," the SQL part could execute the SQL command shown in Figure 9.

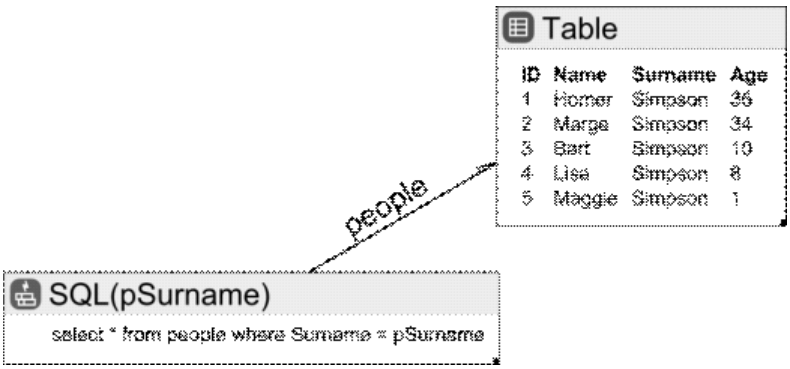

Figure 9. An SQL part that queries a table part.

The SQL part can also be wired to data parts and code parts for use in the SQL command as shown in Figure 10.

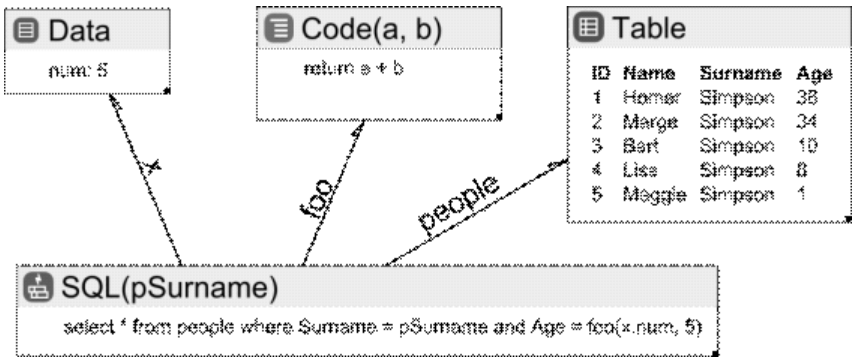

Figure 10. An SQL part references a data part and a code part in the SQL command to query a table.

An SQL part can also be linked to other SQL parts to create nested queries. In this case, the nested SQL part can be referenced as if it were a table. The syntax also allows parameter passing. For example, if a second SQL part takes a parameter "pMaxAge" and is wired to the SQL part in Figure 9 with the alias "query," it could execute the SQL command shown in Figure 11.

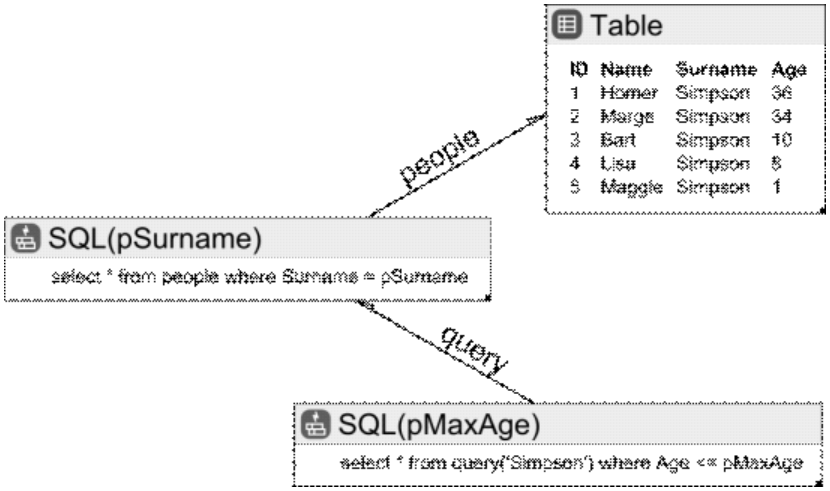

Figure 11. An SQL part references another SQL part to perform a nested query.

In this example, the "query('Simpson')" portion of the SQL command will resolve to the result of the previous query passing the surname "Simpson" as a parameter. The net result of the second query is to return all records with the surname "Simpson" and whose age is less than or equal to pMaxAge.

If a code part is wired to an SQL part, it can execute the SQL command by calling the alias of the wire as a function with parameters as shown in Figure 12.

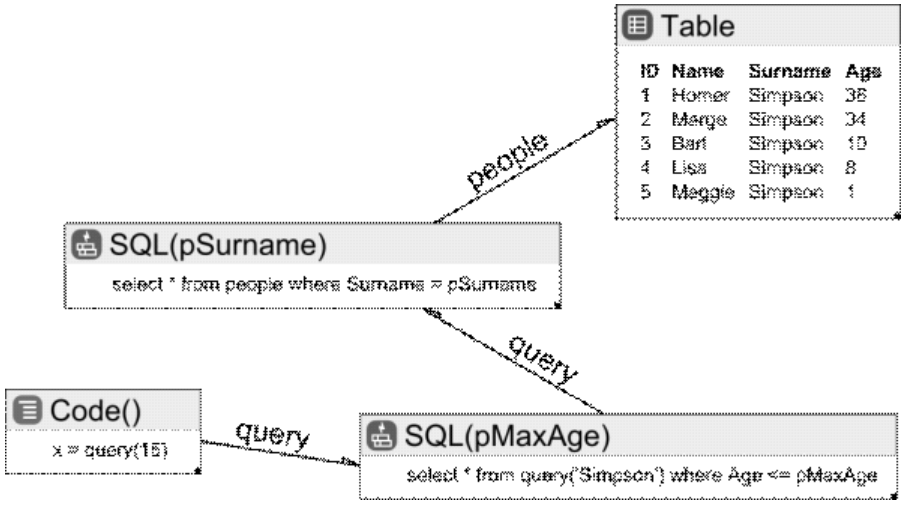

Figure 12. A code part calls an SQL part as a function storing the result to a variable.

Calling an SQL part as a function from a code part will return the query result (if any) as a sheet object with named columns matching the fields of the query result.

2.1.7 Clock Part

The clock part is a customizable clock that is used to set the time parameter when calling the signal function from a code part to schedule a future event. Thus, any specified time or time delay defined in the simulation must be associated with a particular clock part. The current time (in clock ticks) and the tick rate of the clock can be set at any time. Additionally, the current calendar date (year, month, day) and time (hour, minute, second) and calendar unit (e.g. one clock tick = one day), can be set at any time.

The clock part behaves like a real wall clock in the sense that it can be set to any time the user wants and the clock will tick forward from that time as global time advances. This allows different sub-models to use different clocks within the same simulation. For example, if one sub-model is built assuming that one clock tick corresponds to 1 day, and a second sub-model is built assuming that one clock tick corresponds to 1 week, these sub-models can be run together simply by setting the speed of the first clock to 7 so that it ticks 7 times for each single tick of the weekly clock.

To use a clock part to set the time parameter in a signal function as shown in Figure 4, the clock part is called with a single parameter corresponding to a time or time interval on that clock. Assuming a code part is wired to a clock part with alias “clk” and another code part with alias “foo”, Table 3 lists the supported call types for scheduling “foo” to execute at a future time on the clock.

Table 3. List of call types when specifying the time parameter in the signal function using a clock part.

Clock call type	Meaning
signal(foo, (), 1, clk(5))	Execute “foo” at time 5 according to “clk”
signal(foo, (), 1, clk.delay(10))	Execute “foo” after 10 time units according to “clk”
signal(foo, (), 1, clk(2012, 1, 1))	Execute “foo” at 12:00 am on January 1 of 2012 according to “clk”
signal(foo, (), 1, clk(2012, 1, 1, 14, 30, 15))	Execute “foo” at 14:30:15 on January 1 of 2012 according to “clk”
signal(foo, (), 1, clk.delay(months = 2, days = 10))	Execute “foo” after 2 months and 10 days according to “clk”

When the clock part is called with a time or time interval parameter, it computes a global time at which the event will be executed so that all pending events in the simulation can be sorted according their global time. However, the user only ever interacts with times according to specific clock parts.

2.1.8 Hub Part

The hub part provides access to multiple other parts that the hub is wired to. A reference to each part that the hub part is wired to is stored as a member variable in a container class where the name of each member variable is the alias of the wire. Hub parts are used to provide easy access to a group of related parts. The hub part is represented in the GUI as a ring to which wires may be connected. The hub part’s name and other information are not displayed. For example, if the hub part is wired to a data part with alias “x” and a code part with alias “foo,” then a code part that is wired to the hub with alias “h” can access “x” and “foo” using dot notation as “h.x” and “h.foo” as shown in Figure 13.

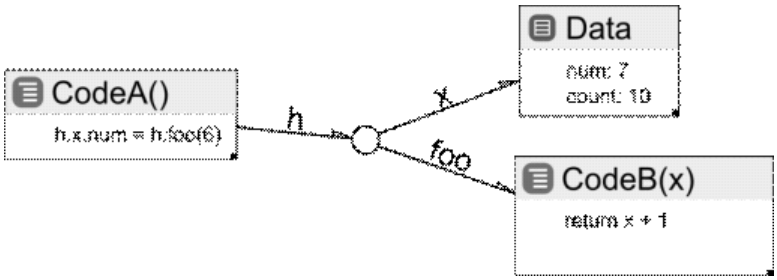

Figure 13. A code part accesses multiple other parts through a hub part.

2.1.9 Actor Part

The actor part is a container for a group of related parts. Actor parts are passive in that creating an actor around a group of parts or deleting an actor has no effect on the operation of the other parts. An example of an actor and its contents are shown if Figure 14.

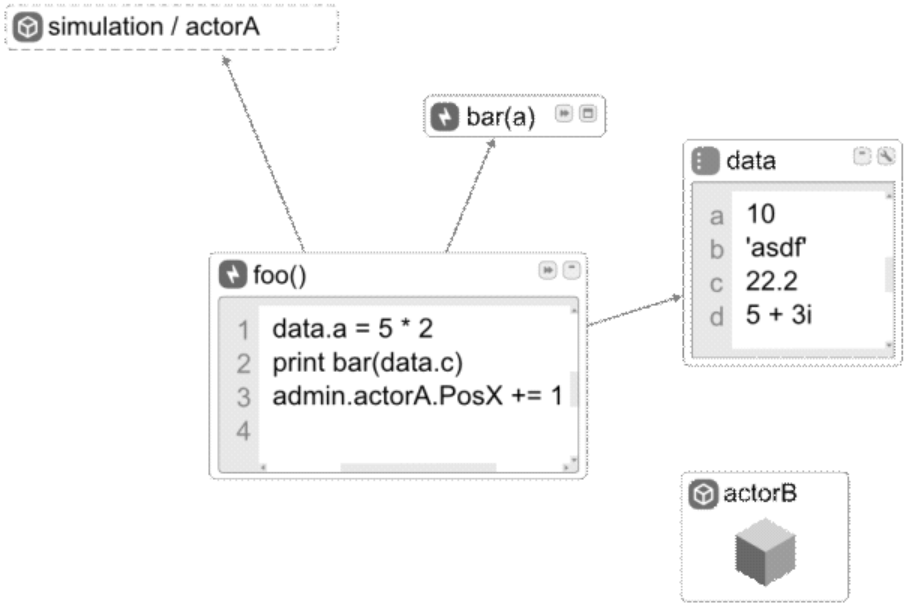

Figure 14. An actor part containing simulation logic defined by various other parts including another actor.

Actor parts are used to hide implementation detail and to organize a model hierarchically. The GUI always displays the contents of a single actor (for example, actorA in Figure 14). This parent actor is shown with a dashed border with a label that displays the path through the actor hierarchy from the top-level simulation actor to the currently viewed actor. Parts can link to their parent actor as shown in the figure. Linking to the parent might be used to change the parent actor's icon or to change its position (as it appears within its parent). In the figure the code "admin.actorA.PosX += 1" would result in moving the parent actor one unit along the x-axis within the simulation actor's 3D space.

Actors can contain sub-actors, such as actorB in the figure, in which case the actor is represented by a 2D image or 3D model icon. Through the GUI, the user can choose to "go to" actorB, and the view would then switch to display the contents inside that actor, or the user could choose to go back up to the parent actor.

Actor icons can be chosen by the user by selecting from a set of default images and 3D models that are included with the application. The user may also select an image or 3D model file from their hard drive, or download a 3D model from Google's 3D warehouse.

2.1.10 Node Part

The node part is a passive part that points to another part that it is wired to. Node parts are used to break wires into segments, 1) for alternate arrangements of wires for better visual effects, and 2) for the functionalities of sockets (explained below). They can only be wired to a single other part, but multiple parts can be wired to a node part. Node parts are represented in the GUI as a

small solid circle. The node part's name and other information are not displayed. The alias of a wire connecting the node part to another part is not used and is therefore not displayed. For example, if a node part is wired to a data part containing variables "num" and "count," then a code part that is wired to the node part with alias "x" can access the data part as if it were directly wired to it as shown in Figure 15.

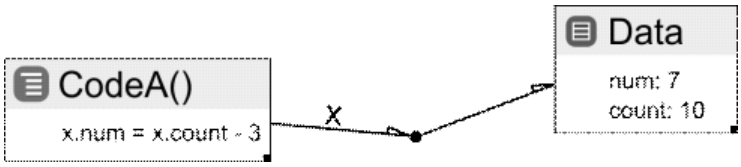

Figure 15. A code part is wired to a node part which is in turn wired to a data part allowing the code part to access the data part as if it were directly wired to it.

Multiple node parts may be wired together to form a network. Within a node network, the direction of wiring (i.e. node A to node B or node B to node A) does not matter, so directional arrowheads are not shown on wires between nodes. Collectively a node network behaves like a single node. Thus, only a single wire can connect from the node network to another part (the network exit wire), but multiple wires can connect other parts to the network (network entry wires). Each part that has a network entry wire has access to the single part that the network exit wire is connected to. An example of a simple network of three nodes is shown in Figure 16.

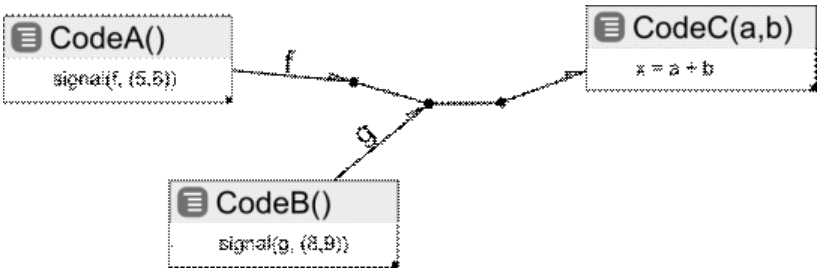

Figure 16. A chain of nodes is wired to a code part (CodeC). Two other code parts (CodeA and CodeB) that are also wired to the chain can access CodeC, signalling it with parameters.

If a single node part or network of nodes is not wired to another part, the node parts all reference the Python 2.7 None value. For example, if a code part is wired to a node part with alias "x," but the node part is not wired to another part, the alias "x" will reference None.

2.1.11 Socket Part

The socket part is a container that groups related node parts. The nodes in the socket are aligned vertically or horizontally. The socket provides functions to add and remove nodes. An example of a socket containing three nodes is shown in Figure 17.

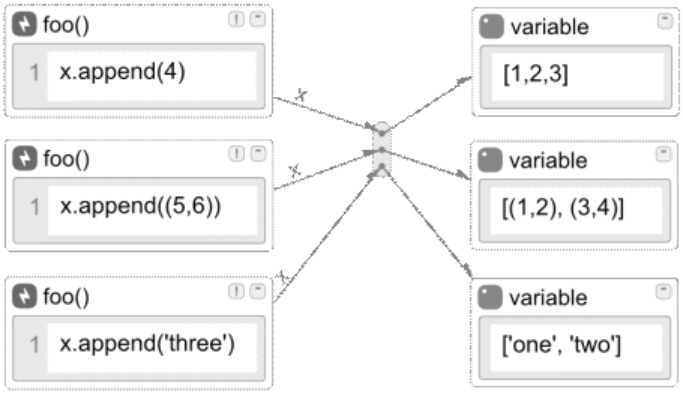

Figure 17. A socket part containing three node parts through which various code parts are accessing variable parts.

In addition to being placed within an actor like any other part as in the figure above, sockets may also be placed on the top, left, right or bottom boundary of an actor. This allows the nodes within the socket to cross the actor boundary. The nodes in a boundary socket can be connected to from both inside and outside the actor. This allows parts that are outside an actor to interact with parts that are inside an actor and vice versa. Figure 18 shows an example of an actor with a three-node socket on its left boundary and a three-node socket on its right boundary. As shown in the figure, boundary sockets are displayed with a rounded side and a straight side with a dashed line. The dashed line represents the actor boundary, therefore, a socket on the actor's left boundary will appear to the left of its dashed line, and similarly for sockets on the actor's right, top and bottom boundaries.

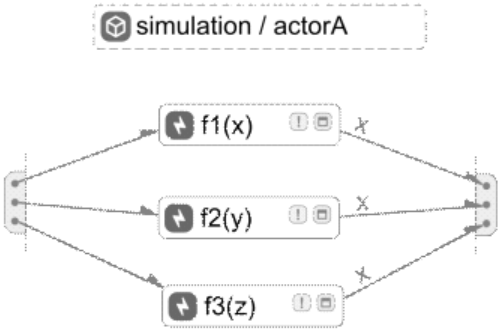

Figure 18. An actor with three-node sockets on its left and right boundaries. Each node on the left socket connects to a code part, and each code part in turn connects to a node on the right socket.

When the actor in Figure 18 is viewed from the outside, the two boundary sockets are visible on the left and right sides of the actor's frame as shown in Figure 19. The nodes in these boundary sockets can then be connected to from outside the actor. Boundary sockets also allow actors to be easily connected together as in the connections between actorA and actorB in the figure.

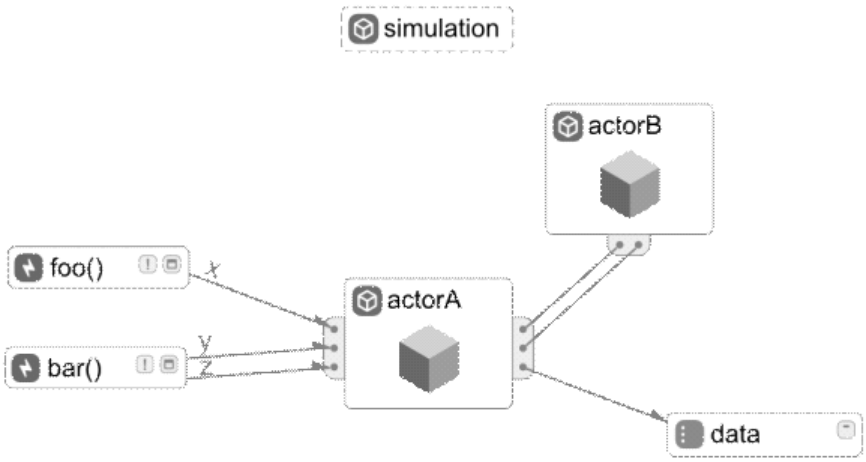

Figure 19. An actor part (ActorB) is wired to other parts through a boundary socket containing two nodes.

2.1.12 Button Part

The button part is a user interaction part that sends signals when the user clicks on it. The button part uses two icons, one is displayed when the button is not pressed, the other is displayed when the button is pressed using the mouse left-click. Both icons may be 2D images or 3D geometry.

In order for the button part to trigger events, it must be wired to one or more code parts. The default behaviour for a button is a push-button (i.e. the button state matches the mouse click state), however, the button part can optionally be configured to behave like a toggle switch where the button state toggles on each mouse down-click event.

When the user left-clicks on the button part, it will immediately call any code parts that it is wired to. The code parts that the button is wired to may be parameterless. But if they do accept a parameter, the calling button will pass a Boolean value indicating whether a down-click (True) or up-click (False) event has occurred. When the user releases the left-click, the button will again call any linked code parts. If the user clicks down on the button part and drags the mouse pointer off the button part without releasing the mouse button, an up-click event will be triggered.

The button part can also be configured to call any linked code parts on down press events only, or release events only. Wires connecting from button parts do not have their name information displayed. An example of using a button part is shown in Figure 20.

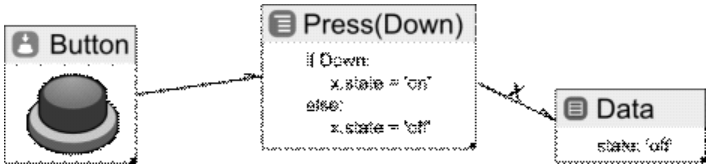

Figure 20. A button part is wired to a code part that sets a state variable in a data part when the button is pressed and released.

A code part that is wired to a button part can access a Boolean state property (e.g. Button.state) to determine if the button is in the pressed or un-pressed state.

2.1.13 Multiplier Part

The multiplier part can be wired to any part that can be called as a function; this includes code parts, SQL parts and nodes that are linked to code parts or SQL parts. The linked parts must all accept the same number of parameters. When the multiplier itself is called or signalled (possibly including parameters) by another part, it will call every part that it is wired to passing the incoming parameters in each call. An example of signalling a multiplier is shown in Figure 21. The purpose of the multiplier is to provide an easy way to call a large number of parts at once, for example, to send a reset signal throughout a simulation. There are no restrictions on the number of wires that can go into or go out of a multiplier part.

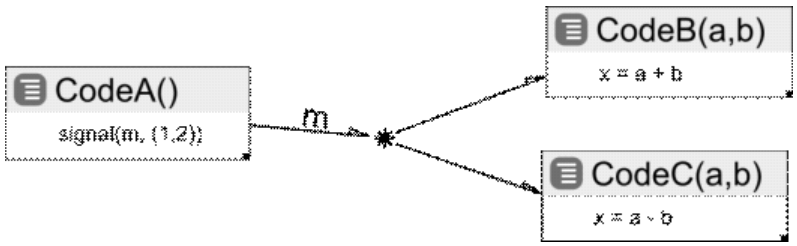

Figure 21. A code part (CodeA) signals a multiplier part passing (1, 2) as parameters; when the signal is processed the multiplier part will call CodeB and CodeC with the parameters (1, 2).

2.1.14 Output Part

The output part is a part that displays charts. Other parts that are wired to the output part provide it with the information to be plotted on the chart. The output part provides the functions shown in Table 4 that allow other parts to set the information to be displayed on the chart:

Table 4. List of functions available with an output part.

Function	Description
set_title(title)	Sets the chart title.
set_xlabel(label)	Sets the x-axis label.
set_ylabel(label)	Sets the y-axis label.
set_zlabel(label)	Sets the z-axis label for 3D plots.
set_categories(categories)	For bar and pie charts, sets the category labels. The categories parameter must be a list of names, for example [‘wins’, ‘losses’, ‘draws’].
add_data_series(name, type,	Adds a data series to be plotted on the chart. The name

data)	parameter is the name of the data series which will be used in the legend. The type parameter may be one of ‘bar’, ‘line’, ‘pie’, ‘xy’, ‘xyz’ or other standard chart types. Other types will be defined in future builds. Some types can be overlaid upon each other, for example a line plot on top of a bar chart. The data parameter contains the data to be plotted. In most cases, this is a list of numbers. For xy and xyz plots, this must be a list of tuples containing the 2D or 3D coordinates of the data points to be plotted.

Additional functions must be provided that can be used to configure all the graphical settings of the chart including gridlines, legends, axis numbers, axis tick marks, axis scaling, etc. Automatic calculation of the axis scales must be done by default.

The output part updates the chart display each time one of the above functions is called that modifies the information to be displayed.

An example of an output part whose chart information is set by a code part is shown in Figure 22.

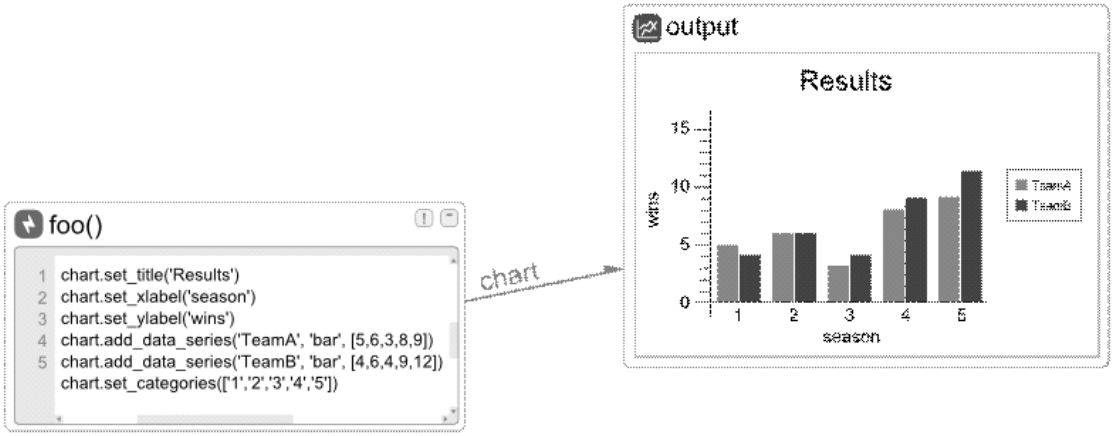

Figure 22. An output part displays a chart that is created by a code part that links to the output part.

2.1.15 Comment part

The comment part displays arbitrary information or commentary that may be used for any reason, usually to assist with understanding how a model works.

Comment parts are displayed with a unique simplified visual style to distinguish them from the rest of the model logic. They can be wired to other parts to show what the comment refers to if desired, but this serves only a cosmetic purpose. If another part is wired to a comment part, it can set the comment string that is displayed in the comment part.

Figure 23 shows an example of a comment part being used to describe another part in the model.

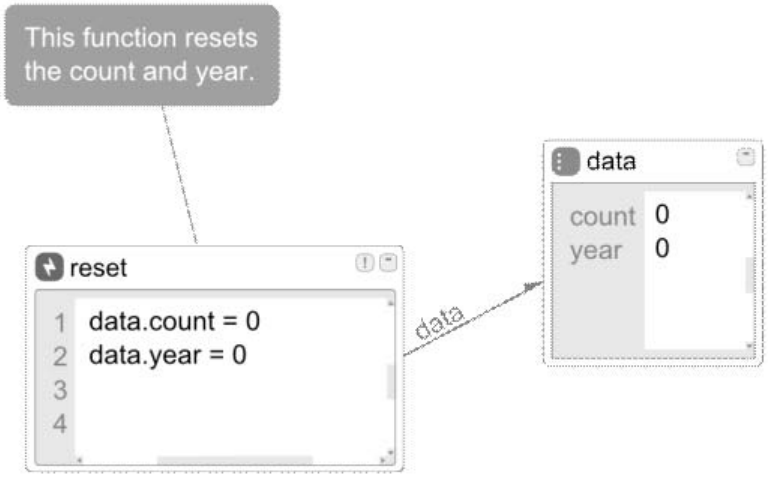

Figure 23. A comment part describing the action of a function in a model.

2.2 Simulation Execution

When a code part is signalled, its function, parameters, priority and global time are stored in a global event queue. When an event finishes processing, the simulation engine selects the next event to process from the event queue. To pick the next event, it selects the event(s) with the soonest time, if there are multiple such events, it selects from among them the highest priority event(s), if there are still multiple such events, it selects from among them the event that has been in the queue the longest (First In First Out (FIFO) execution). The R4 HR application also provides a global option to use Last In First Out (LIFO) execution. In this case, where there are multiple events with the same time and priority, the event that was most recently added to the queue is selected. Once the event has been selected, it is removed from the global event queue, the simulation's global time is advanced to the time of the event and the simulation engine executes the event function passing it the event parameters.

The simulation is either in a "pause" state, a "run (interactive)" state, a "run (real-time)" state, or a "run (non-interactive)" state. In the pause state, the simulation state is frozen and the simulation engine does not retrieve events from the event queue. In this mode, the user can carry out model building actions in the GUI such as creating, manipulating and interconnecting parts. The user can manually step the simulation forward in the pause state. The step will retrieve and process the next event on the global event queue.

In the run (interactive) state, the simulation engine will continuously process events off of the event queue, and it will continuously refresh the simulation display in the GUI so that any animated effects such as the movement of parts and updating part displays will be observed. The simulation will stay in this state even if there are no events on the queue and will wait for an event to occur. In the run (real time) state, the simulation engine behaves identically to the run (interactive) state except that the processing of events is aligned with a scaled version of real time. The user is able to select the scale to be applied so that, for example, one minute of real time corresponds to a single tick on the master clock. In this mode, delays in the model will correspond to real time delays as the simulation executes. Finally, in the run (non-interactive) state, the simulation engine continuously processes events off the event queue until there are no events remaining. However, it does not refresh the GUI display. This mode will typically be used for performance reasons once a model has been tested and the user wishes to perform a

simulation as quickly as possible. It is possible to switch between these states at any time through the GUI or within a code part by calling a function that sets the simulation state.

When running in non-interactive mode, multiple replications of a given scenario can be executed one after the other or in parallel on systems with multiple CPU cores. Replications are identical instances of a simulation scenario that differ only due to stochastic effects during execution. For example, an event may be programmed to occur at a specified time with a 50% probability. Therefore, in some replications, the event will occur; in others, it will not.

3.0 3D GRAPHICAL USER INTERFACE

This section describes the features of the graphical user interface (GUI). GUI functions will be available as appropriate through a main menu, short-cut toolbars, clicking, dragging and dropping model objects, right-click context-sensitive menus, and other keyboard and mouse action. In general, global actions (e.g. saving/loading a scenario) are accessible in the main menu while object-specific actions (e.g. editing a part) are accessed by right clicking on the relevant object and selecting the action from the context menu.

3.1 General Features of the GUI

The main menu will contain a file menu with options to create a new scenario, load a scenario, save a scenario, save a scenario as a new file, export selected parts and import parts from an existing saved file. It will contain a view menu to adjust various options that affect how the GUI displays the current simulation scenario such as showing/hiding optional information, adjusting fonts, etc. The view menu will also contain options for navigating and searching within the current scenario. It will contain a simulation menu that lets the user set the simulation execution state, view and edit the global event queue, and manage the use of available CPU resources. It will contain a tools menu that will include various utilities including a user preferences dialog. Finally it will contain a help menu providing various help resources to the user. Additional menus and menu items will be added in future builds.

In the main modelling window, the user is presented with the view inside of a single actor showing the parts it contains and the wires between those parts as shown above in Figure 19. This includes sockets and nodes that are on the boundary of the actor. The actor being viewed is termed the “current actor”. Each scenario has a root simulation actor that is the top-level parent for all parts in the scenario. When users start a new scenario, they create their model inside the root actor.

The current actor can be changed to any sub-actor part within the current actor by double-clicking the sub-actor or right-clicking the sub-actor and selecting “go to” from the context menu. Alternatively, the view can be changed to the current actor’s parent by right clicking empty space and choosing “go to parent.” Additionally, an “actor hierarchy” dialog can be launched from the view menu that allows the user to change the current actor to an arbitrary actor by selecting it from a tree view of all the actor parts in the current scenario.

Except for nodes, sockets, hubs and multipliers that use specific graphics in the GUI, the contents of all parts are displayed within the frame below the title bar. If the contents of the part become too large for the frame, vertical and horizontal scroll bars will appear allowing the user to scroll to the contents they wish to see.

To hide certain details from view or to emphasise certain parts of a model, the standard part view can be adjusted in a number of ways. Figure 24 shows an example of the various view options available on a typical part.

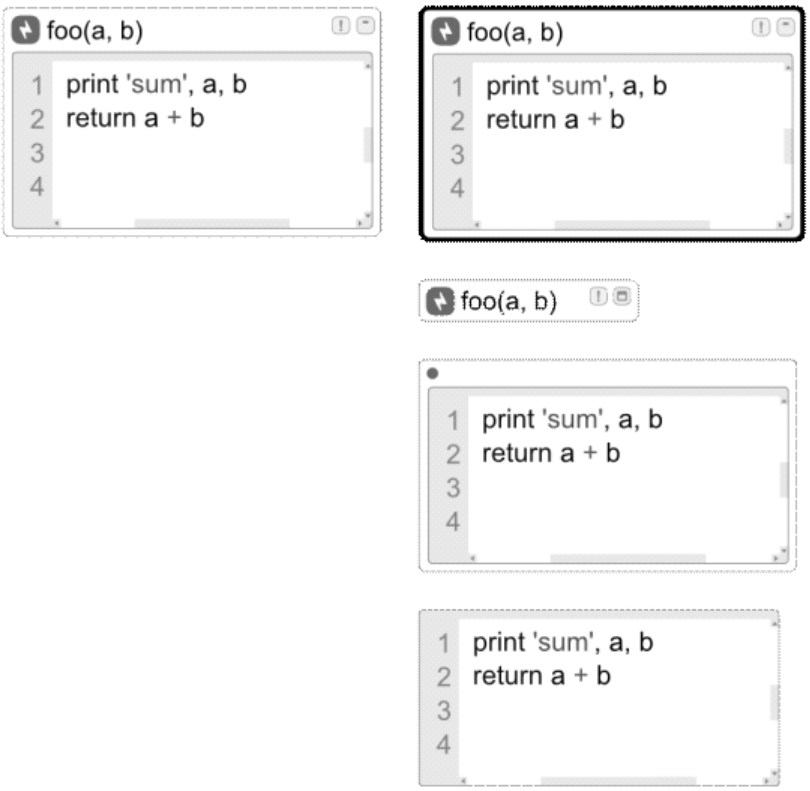

Figure 24. Part view options showing (clockwise from upper left) a typical code part in standard full view, bold frame, hidden contents, hidden title, and hidden frame.

Every part in a model can be dragged and dropped with the mouse to change its location. A part can be selected by clicking on it and multiple parts can be selected by holding control and clicking many parts or by holding shift and dragging a box over the desired parts. Once one or more parts are selected they can be moved by dragging, they can be cut/copied and pasted to a new location in the model, and they can be deleted. When a part or group of parts is cut/copied and pasted to another location, deep copies of the parts are created, i.e.: modifying a copy of a part does not affect the original. This includes all child parts of an actor if an actor part is copied. All wires between parts being copied are also copied.

When a single part is selected, resize arrows are drawn around the selection highlighted that can be dragged to resize the part as shown in Figure 25.

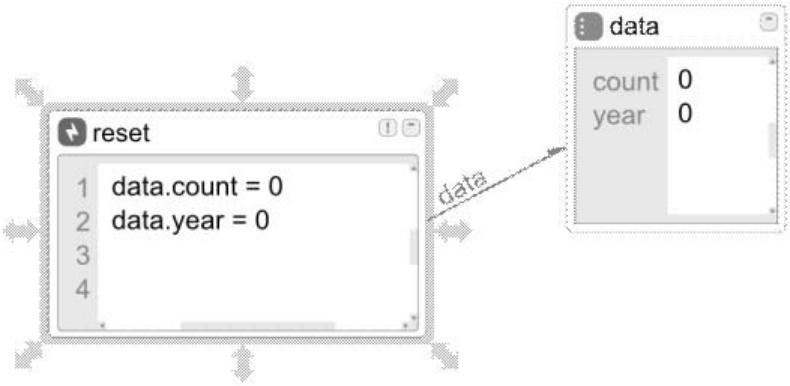

Figure 25. A highlighted part showing resize arrows that can be dragged to resize the part.

When multiple parts are selected, they can also be collapsed into a new actor part. This encloses the selected parts and wires inside a new actor part. For each wire that crosses the new actor boundary, a boundary socket and node is created on the border of the actor allowing the connection to cross the actor boundary. The collapsing of selected parts into an actor has no effect on the functioning of the model. All wires remain intact except that some of them cross the actor boundary through a node and all parts remain except that some have moved inside the new actor.

The user's viewpoint, called the camera, can be moved and oriented in 3D space. The default camera controls are shown in Table 5. Alternate camera control options will be added in future builds.

The application allows comprehensive undo and redo actions with memory of many consecutive actions. This functionality applies to any change made to the model including creating, editing and deleting parts and creating, editing and deleting wires.

Table 5. List of default camera controls.

Camera Control	Action
Mouse wheel up/down	Move camera up/down
Shift + mouse wheel up/down	Move camera left/right
Control + mouse wheel up/down	Move camera forward/backward (zoom in/out)
Control + shift + mouse wheel up/down	Rotate camera left/right around a vertical pivot axis. If no parts are selected, the pivot axis is located at the centre of all parts in the current actor. If one or more parts are selected, the pivot axis is located at the centre of the selected part(s).

3.2 Model Building

Models are created by adding and editing parts and interconnecting them. New parts are created by right-clicking on empty space which opens a context menu that includes a “new part” option. This opens a sub menu that lists the available part types. When a part type is selected, it is created at the location that the user initially clicked.

To create a wire from one part to another, the user presses a predefined key that toggles on a wiring mode indicated by changing the mouse pointer to an appropriate icon. While in wiring mode, hovering the mouse over a part will place a red highlight around the part. The user then clicks the part from which the wire should originate. This part now remains highlighted. At this point, hovering the mouse over potential destination parts for the wire will highlight the destination part in red, and draw a red wire from the first to the second part. Clicking a destination part confirms the wiring action, creates the wire from the first to the second part and everything that was highlighted in red returns to its original colour.

The new wire name defaults to the name of the destination part. If the source part already has a wire with that name, a number is appended to the wire name to make it unique. Once the wire is created, the wiring mode is deactivated.

Wires can be selected by clicking on them and can be included in multiple selections involving other wires and parts.

The names of all parts and wires can be quickly changed by double-clicking on the name to trigger a text editing mode, typing the new name and pressing enter.

3.3 Part Dialogs

Once a part has been created the user can interact with it to set its contents or behaviour. Every action that can be performed on a part will be accessible through a context menu that appears when the user right clicks on the part. Some elements in the context menu will be common to all parts including “cut”, “copy”, “delete”, “properties…”, “edit…”, “interact’, and “help.” Some elements in the context menu will depend on the type of part that is clicked on. For example, the context menu for a code part includes a “run” option that runs the code. The items in the context menu may have shortcuts available in various forms. These include keyboard shortcuts (for example, control-c will copy the part), and shortcut buttons that appear in the upper right corner of the part (for example, code parts have a shortcut button that runs the code).

The properties context option opens a dialog that allows the user to set standard attributes relevant to all parts including the name, 3D position and hidden status.

There are two methods through which a part can be modified; "interact" and “edit”. The same tasks can be accomplished using either “interact” or “edit”, but “interact” allows the users to manipulate the part from a command prompt, whereas “edit” allows the users to manipulate the part in a user-friendly graphical interface.

“Interact” launches an interactive console like the traditional Disk Operating System (DOS) prompt to interact with the part, wherein the user enters commands to edit the part. For example, if the user selects to interact with a data part named SimData, the user could then type the following into the interactive console “SimData.population = 200.” This action would have the same effect as if a code part was wired to the data part (with the alias SimData) and this line was executed from the code part. “Interact” is an advanced feature that will be required by a limited number of users.

"Edit" launches a graphical interface window to edit the part. Double clicking a part automatically launches the part editing dialog (the same action as selecting edit from the context menu). Double clicking the part title bar automatically launches the part properties dialog (the same action as selecting properties from the context menu) and automatically highlights the title field so the user can immediately start typing and then press enter to quickly rename a part. On a code part, "edit" would launch the code editing window that lets the user write code in an advanced text editor.

The rest of this section describes the unique actions available in the context menu for each part type, and it describes the functionality of the edit dialog for that part.

Code Part

The code part includes a context menu option to run the code. This action is also available as a shortcut button.

The code part edit dialog will include a field for setting the function parameters and a text editor for writing and editing code. The text editor will have a rich feature-set to support Python 2.7 coding including syntax highlighting, auto indenting, parenthesis matching, code folding, and auto-completion. Links to Python 2.7 resources will also be provided such as a help field that launches a web browser and searches the online Python 2.7 language reference.

Variable Part

The variable part edit dialog provides a string input field that allows the user to set a new value for the variable part. For example, to a create a Python 2.7 list in the variable part, the user would enter the string “[‘one’, ‘two’, ‘three’]” in the input field. The string is then evaluated to create the Python 2.7 object and stored in the variable part.

Data part

The data part edit dialog allows the user to view and edit the current data entries, create new entries and delete entries. To create a new entry, the user will enter a name and a string defining the value to be created. This string will then be evaluated to create the Python 2.7 object to be stored in that entry. It will also allow the user to rearrange the order in which the entries will be displayed by dragging and dropping, and it will allow the entries to be sorted alphabetically or by order of creation.

Sheet Part

The edit context option for the sheet part launches a dialog window allowing the user to view and edit the contents of the sheet in a grid view similar to a spreadsheet program. Clicking on a cell in the grid allows the user to enter the value to be stored in the cell. When the user presses “Enter” after typing the contents of a cell, the entry is evaluated and the resulting Python 2.7 object is stored in the cell. For example, entering "(1,2,3)" would create a three-tuple with elements 1,2,3. This dialog also allows the user to add and remove rows and columns, create named columns and cut/copy and paste the contents of one cell to another cell. The sheet editing dialog also allows the user to make ranged selections by dragging from one cell to another. A selected range can be cut/copied and pasted to another location in the grid.

Table Part

The table part edit dialog allows the user to view and edit the table including adding rows, editing row data, deleting rows, adding fields, deleting fields, changing field names and field order, adding indexes and removing indexes. It will also allow the user to sort the data in ascending or descending order on a selected field and to apply filters to each field to filter the data shown. The table part dialog will allow table data to be imported from and exported to external sources including MS Access tables.

SQL Part

The SQL part edit dialog provides a graphical SQL query design tool that will allow the user to drag and drop tables and queries, where the available tables and queries are those that are linked to by the SQL part. The fields of each table and query will be displayed. The user can then select the type of query and create joins by dragging between fields in different tables/queries and can define the fields, criteria, grouping, ordering, etc to be used by the query.

Clock Part

The clock part edit dialog provides fields to set the time (in ticks), speed, calendar date (year, month, day), calendar time (hour, minute, second), and calendar unit (e.g. 1 tick = 1 day). The clock part context menu will include an option to display the clock time in ticks or calendar time.

Actor Part

The actor part edit dialog provides the ability to set the actor's icon using a 2D image or 3D model file. To choose the icon, the user will be able to select images and 3D geometries in standard file formats from a location on the user's system. A collection of built-in images and 3D models of standard shapes and symbols will be available as "clip art."

Supported file formats for images must include png, jpeg, gif, and bmp. Supported file formats for 3D geometry must include collada (.dae), Google Sketchup (.skp) and Blender (.blend).

Node Part

The node part does not have an edit context option. Double-clicking the node part launches the standard properties dialog.

Socket Part

The socket part context menu includes options to "add node" and "remove node" and to set the boundary side to "none" (if not crossing the actor boundary) or "left", "right", "top" or "bottom" to create a boundary socket.

The socket part does not have an edit context option. Double-clicking a node part in the socket launches the standard properties dialog.

Button Part

The button part edit dialog provides an interface to select the two icons representing the pressed and un-pressed state of the button. The process for selecting both icons is identical to that of the actor part and provides the same features. The dialog also provides the option to turn on or off the press event and the un-press event. For example, if the user only wishes to process the press event, the un-press event can be turned off. The dialog also provides an option to select between push-button and toggle-switch behaviour.

Output Part

The output part context menu provides the option to export the chart as a high-quality image file in pdf and png formats. The user will have the option to select the resolution of the saved image file.

The context menu will also provide the option to export the chart data to a csv text file suitable for importing into other software including MS Excel and MS Access.

The output part edit dialog allows the user to customize all aspects of the chart including the title, axes labels, category labels, axis tick points, grid lines, legends, second y-axes, colors, line thicknesses, etc. The dialog will also display the chart data in tabular form and allow the user to edit the data points, add and delete data points, and add, rearrange and remove data series.

Hub Part

The hub part does not have an edit context option. Double-clicking the hub part launches the standard properties dialog.

Multiplier Part

The multiplier part does not have an edit context option. Double-clicking the multiplier part launches the standard properties dialog.

3.4 Saving and Loading

An open scenario can be saved to a ".ori" file at a user-specified location on their system. One or multiple selected parts in a scenario can also be exported to a .ori file. A .ori file can also be imported into an already open scenario. There is no formatting difference between a .ori file containing an entire scenario and a .ori file containing a single part. A complete scenario saved in a .ori file could be imported into another scenario.

Each .ori file implements a basic form of version control. The .ori file stores a project name and description for the content of the file. When the user saves a scenario or part or group of parts to a .ori file, the save interface allows the user to provide a version number, version name, version description, details of changes since the last version, and description of VV&A (validation, verification and accreditation) status. If the user does not change the version number from an existing version, the save overwrites that version. If the user provides a new version number, the save adds to the file under the new version. When the save occurs, a time stamp is attached to the version information.

When the user opens a .ori file, a "versions" interface is displayed that allows the user to explore all versions stored in the .ori file. It allows the user to view and edit the project name and description and to view and edit the information on each version including the ability to delete versions. The user may then close the versions interface, or open a specific version to begin working on a model. To open a specific version, the user selects that version and clicks open. By default, the last version opened is selected so the user can immediately click open to begin working on that version.

A number of scenarios have been created using the R4 HR prototype. The scenario files are SQLite databases that contain the complete state of a simulation scenario. The R4 HR application must have the ability to open or import these prototype scenarios and execute them. It must also be possible to save the current simulation state back to the prototype file format.

3.5 Executing Simulations

The Simulation menu allows the user to select from the four run states: pause, run (interactive), run (real-time) and run (non-interactive). When the simulation is in the pause state, a "step" option becomes available in the simulation menu. This causes the simulation to execute only the next event on the global event queue.

Except when running in non-interactive mode, it is possible to manually run any code part by clicking the "run" shortcut button or right-clicking and selecting "run" from the context menu. If the code part has parameters, a dialog will open that will allow the user to enter a value for each named parameter. The input parameter strings will be evaluated and then pass to the function.

Except when running in non-interactive mode, an option to "view event queue" is available in the simulation menu that opens an interface to view the entire global event queue. The events will update dynamically as the simulation runs. When paused, the interface allows the user to edit individual events and to delete events.

The run (interactive) and run (real-time) execution modes only run one replication of the scenario because the user can only interact with one scenario at a time. When the user chooses the run (non-interactive) execution mode an interface is displayed that allows the user to set the number of replications to run and, in the case of systems with multiple CPU cores, set the number of replications that can be executed in parallel, i.e. the maximum number of CPU resources that will

be used. Various stopping conditions can also be defined including a maximum simulation time and a maximum real time (e.g. time-out after four hours).

In order to run simulations on high-performance computing servers such as Target Production Environment 3, a “run-time” version of the R4 HR application must be provided that can be invoked from the command line and that will execute non-interactive, multi-replication scenarios. To accomplish this, the user would develop the scenario using the R4 HR GUI on his or her own PC system including setting up the parameters for a non-interactive run. Then the user would submit a job to the high performance server requesting a certain number of CPU and memory resources to execute the scenario. The job script will invoke the run-time version of R4 HR telling it to execute the specified scenario. The run-time version would then run the specified number of scenario replications in parallel on the specified number of CPU resources.

3.6 Event Indicators

When viewing scenarios in the pause, run (interactive) and run (real-time) modes, the GUI displays event indicators showing which code parts and which actors have pending events. The event indicators are displayed above the upper left corner of the code parts and actor parts. There are three event indicators as follows:

1. A green circle with an arrow is the next event indicator and appears on the code part that is the next event on the global event queue. There can only be one code part with this indicator. This indicator also appears on all parent actors of the code part so that the user can easily find the code part that will execute next.

2. An amber circle with a number is the concurrent event indicator and appears on code parts that are scheduled to execute at the same time as the next event, but which will execute after the next event according to the sorting of the global event queue. The number within the circle indicates the number of concurrent events that are pending on the code part. Actors display the total number of concurrent events among code parts that are below it in the part hierarchy.

3. A red circle with a number is the waiting event indicator and appears on code parts that are scheduled to execute at a time later than the next event. The number within the circle indicates the number of waiting events that are pending on the code part. Actors display the total number of waiting events among code parts that are below it in the part hierarchy.

The user can double-click on any of the event indicators to open an interface to view and edit the pending events on a code part or actor part. This interface will allow events to be created, edited (to change the parameters, time or priority) and deleted. The option to delete all events on a code part or actor part is also available.

Figure 26 provides an example of the event indicators and illustrates their appearance.

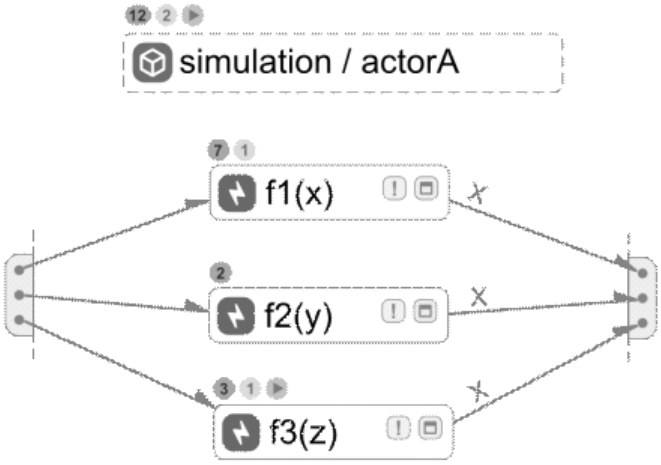

Figure 26. An actor part contains three code parts that are scheduled to execute on the event queue illustrating the display of event indicators.

3.7 Errors, Debugging and Help

If an error is encountered while compiling a code part or while executing a code part, the simulation is paused (if not already paused) and a warning shortcut button appears on the code block. The line of code at which the error occurred is highlighted in red for a compilation error, or in amber for a runtime error. Clicking on the warning shortcut button displays the error message.

To assist with debugging, the user can set breakpoints on code parts and on specific lines of code within code parts. A breakpoint on a code part pauses the simulation when an event on this code part becomes the next event in the global event queue.

A breakpoint on a specific line in a code part, causes the simulation to enter a debugging mode when this line of code is reached allowing the user to advance through the code line-by-line and to investigate the values of any local variables used in the code part.

If the user right-clicks on a code part, a “debug” option is available that commences execution of the code and initiates line-by-line debugging at the first line allowing the user to manually step through the code.

During line-by-line debugging, if the user steps past the last line of code in a code part, line-by-line debugging continues at the first line of the next code part to execute.

Access to resources to assist with building models and writing code for code parts will be available in the help menu in the form of a user manual, part documentation, tutorials, Python 2.7 reference documents, working code examples and working sample models. Help resources should be readily searchable from the modelling GUI. For example, highlighting a piece of code such as a function name, right-clicking the highlighted function and selecting “help” will display relevant documentation.

The tools menu will provide a “find” capability to search the entire simulation (including part names, wire aliases, and part contents) for a specified keyword or keywords.

3.8 Event Logging

As the simulation executes, the user has the option to record events to a global event log. The user can specify automatic logging of events in which case the level of detail reported in the log can be adjusting from minimal (errors) to moderate (errors and warnings) to extensive (errors, warnings, and events). The user can also write custom messages to the log using a “log” function in a code part. The user can choose to view the log from the tools menu and can save the log to a file.

3.10 Other Requirements

Other requirements for the R4 HR application include:

a) Python 2.7 must be the language used in code parts.

b) Python 2.7 must be the language used to develop the R4 HR simulation environment (C can be used to develop performance-critical sections of code to be compiled into Python 2.7 modules and included in the main Python 2.7 project).

c) Any additional software components incorporated into the R4 HR software must be approved by the Technical Authority.

APPENDIX 5 TO ANNEX A
R4 HR APPLICATION PERFORMANCE SPECIFICATIONS

1.0 INTRODUCTION

The R4 HR application must meet the "Pass" performance on Target 2 as defined in the Target Production Environment (section 5.0 of the SOW).

2.0 SIMULATION ENGINE MEASURES

Measure SE-1: The tool executes model/simulation code correctly.	**Evaluation:** Target: Yes Pass: Yes
Evaluation: 1. The steps of execution follow logically from the simulation scenario definition. 2. The steps of execution are repeatable. That is, tool execution of model/simulation code is successfully validated by comparing execution of models/code built in the tool against execution of logically equivalent models/code built in currently used COTS and the R4 HR prototype.	

Measure SE-2: The tool allows simulations of realistic scenario sizes (similar scale to DND/CF wide integrated models) to be executed in a reasonable amount of time on Target 2.	**Evaluation:** Target: 2 Pass: 2
Evaluation: For each scenario size and hardware configuration tested, the following scale will be used. The individual scores will be weighted by the complexity of the combinations to achieve an overall score. 0 – unacceptable length of time 1 – on the order of days (acceptable in special cases) 2 – on the order of hours (reasonable) 3 – on the order of minutes (ideal)	

Measure SE-3:	**Evaluation:**
The speed at which the simulation engine executes various test scenarios relative to logically equivalent scenarios executed in existing COTS software (Arena and Simio). The scenarios will be simple models that test specific performance characteristics repeatedly. For each scenario, single-CPU and multiple-CPU (parallel) configurations will be measured. The scenarios will include, but will not necessarily be limited to, tests of the following: a. simple variable reads/writes b. database table reads/writes c. signal branching (Arena decide blocks) d. few concurrent signals (approx. 10) e. many concurrent signals (approx. 100,000) f. small scripts (<10 lines of code) g. large scripts (100s of lines of code)	Target: *s* >= 2.0 Pass: *s* >= 0.75

Evaluation: The following scale will be used.

For this evaluation, a 12 core workstation (Target 2) will be used, and 96 replications of each scenario will be executed. Arena only executes on a single CPU core, therefore Arena's execution speed will be compared to the R4 HR application operating in a single core mode. For multi-core execution speed testing, the R4 HR application will be compared to Simio. Simio is a similar product to Arena, but it takes advantage of multiple CPU cores. The scale used to the measure the performance of the R4 HR application compared to the COTS software will be the speed ratio, s:

,
where T_{COTS} is the execution time in Arena or Simio, and T_{R4HR} is the execution time in the R4 HR application.

For each of SE-3a through SE-3g (and possibly others), the R4 HR application will be evaluated by determining the speed ratio for the following comparisons:

i. R4 HR v. Arena, single core
ii. R4 HR v. Simio, single core
iii. R4 HR v. Simio, 4 cores
iv. R4 HR v. Simio, 12 cores

3.0 GUI MEASURES – MODEL BUILDING FUNCTIONS

Measure MG-1: Aesthetics of the interface, including aesthetics of the application window(s) and aesthetics of model flow diagrams built in the interface	**Evaluation:** Target: 4 Pass: 3
<u>Evaluation:</u> 0 – Unacceptable (confusing, illogical, greatly impedes functionality) 1 – Poor 2 – Average (somewhat unclear and cluttered, but does not negatively impact functionality) 3 – Good 4 – Excellent (clean, clear, uncluttered appearance that enhances functionality)	

Measure MG-2: Level of intuitiveness of the interface structure.	**Evaluation:** Target: 4 Pass: 3
<u>Evaluation</u>: 0 – Unacceptable (necessary features are missing and/or hard to find and do not follow standard conventions) 1 – Poor 2 – Average (necessary features in place, some are not easily found, some elements do not follow standard conventions) 3 – Good 4 – Excellent (all necessary features in place, can be easily found, and follow standard design conventions)	

Measure MG-3: Ease/efficiency of performing the following model-building functions: a. creating new blocks b. finding pre-existing blocks c. modifying pre-existing blocks d. connecting blocks to each other (i.e. creating models) e. connecting models to each other (i.e. model integration) f. model verification "Block" is a general term referring to Actors, Arena-like blocks, parts and code.	**Evaluation:** Target: 4 Pass: 3
<u>Evaluation</u>: Each sub-measure (a through f) is rated on the following scale. 0 – Unacceptable (process is illogical, not intuitive, complex, slow) 1 – Poor 2 – Average (process is logical, mostly intuitive, sometimes complex, sometimes slow) 3 – Good 4 – Excellent (process is logical, intuitive, straightforward, fast)	

Measure MG-4: There is a one-to-one correspondence between the graphical representation of a model in the GUI and the scenario file representation of the model in the SQLite database file. a. data (variables, attributes, etc) integrity b. visual display integrity	**Evaluation:** Target: 3 Pass: 3
Evaluation: 0 – Unacceptable (no one-to-one correspondence) 1 – Poor (limited level of one-to-one correspondence; impossible/hard to work with) 2 – Good (good level of one-to-one correspondence; some minor issues that can be managed) 3 – Excellent (perfect one-to-one correspondence)	

Measure MG-5: Utility of debugging features including: a. ease of finding, activating, deactivating debugging functions b. ease of finding errors c. level of clarity of error messages	**Evaluation:** Target: 4 Pass: 3
Evaluation: 0 – Unacceptable (many difficulties in finding and using features, errors not clearly identified, error messages unclear) 1 – Poor 2 – Average (some difficulties in finding and using features, some errors not clearly identified, some error messages unclear) 3 – Good 4 – Excellent (all functions easily found and used, all errors clearly identified, all error messages are very clear and informative)	

Measure MG-6: Utility of the “Help” features. a. ease of finding the desired information b. clarity and usefulness of the information provided c. usefulness of the examples provided d. effectiveness of context-sensitive help text	**Evaluation:** Target: 4 Pass: 3
Evaluation: 0 – Unacceptable (information is hard to find and unclear, examples are incomplete and/or irrelevant, context sensitive help usually does not direct user to relevant information) 1 – Poor 2 – Average (information is easily found, some information lacking in clarity and usefulness, some examples are incomplete, context sensitive help usually directs user to relevant information) 3 – Good 4 – Excellent (information is easily found, very clear and very useful, examples are relevant and complete, context sensitive help always directs user to relevant information)	

4.0 GUI MEASURES – REPORTING FEATURES

Measure RG-1: Aesthetics of GUI reporting features including aesthetics of user interfaces for generating reports and aesthetics of the reports generated.	**Evaluation:** Target: 4 Pass: 3
<u>Evaluation</u>: 0 – Unacceptable (confusing, illogical, greatly impedes functionality, reports are difficult to interpret) 1 – Poor 2 – Average (somewhat unclear and cluttered, but does not negatively impact functionality, reports are interpretable with minor effort) 3 – Good 4 – Excellent (clean, clear, uncluttered appearance that enhances functionality, reports are very easy to interpret)	

Measure RG-2: Level of intuitiveness of the interface structure.	**Evaluation:** Target: 4 Pass: 3
<u>Evaluation</u>: 0 – Unacceptable (necessary features are missing and/or hard to find and do not follow standard conventions) 1 – Poor 2 – Average (necessary features in place, some are not easily found, some elements do not follow standard conventions) 3 – Good 4 – Excellent (all necessary features in place, can be easily found, and follow standard design conventions)	

Measure RG-3: Utility and variety of reporting features: a. ease of generating and working with reports b. variety of types of reports available (e.g. 2D/3D plots, standard charts, tables, statistical plots, etc.)	**Evaluation:** Target: 4 Pass: 3
Evaluation: 0 – Unacceptable (reports not easily generated, very limited manipulation and customization capabilities, many report types lacking) 1 – Poor 2 – Average (reports easily generated and manipulated, limited customization options, lacks some report types) 3 – Good 4 – Excellent (reports easily generated and manipulated, highly customizable, comprehensive set of report types available)	

Measure RG-4: Utility of the “Help” features. a. ease of finding the desired information b. clarity and usefulness of the information provided c. usefulness of the examples provided d. effectiveness of context-sensitive help text	**Evaluation:** Target: 4 Pass: 3
Evaluation: 0 – Unacceptable (information is hard to find and unclear, examples are incomplete and/or irrelevant, context sensitive help usually does not direct user to relevant information) 1 – Poor 2 – Average (information is easily found, some information lacking in clarity and usefulness, some examples are incomplete, context sensitive help usually directs user to relevant information) 3 – Good 4 – Excellent (information is easily found, very clear and very useful, examples are relevant and complete, context sensitive help always directs user to relevant information)	

APPENDIX 6 TO ANNEX A
LEXICON / ABBREVIATIONS

CDR	Critical Design Review
CDRL	Contract Data Requirements List
CF	Canadian Forces
COTS	Commercial-off-the-Shelf
DID	Data Item Descriptions
DND	Department of National Defence
DRDC	Defence Research and Development Canada
GFE	Government Furnished Equipment
GIDEP	Government / Industry Data Exchange Program
GUI	Graphical User Interface
HR	Human Resource
MG	Modelling GUI
MS	Microsoft
OR	Operational Research
PDR	Preliminary Design Review
PRM	Progress Review Meeting
R4 HR TDP	Right person, Right qualifications, Right place, Right time, Human Resources Technology Demonstration Project
RG	Reporting GUI
SDR	System Design Review
SE	Simulation Engine
SME	Subject Matter Expert
SOW	Statement of Work
SQL	Structured Query Language
SRR	System Requirements Review
TRR	Test Readiness Review
VV & A	Validation, Verification and Accreditation
WBS	Work Breakdown Structure

APPENDIX 7 TO ANNEX A

DATA DELIVERABLES

1.0Introduction

1.1Purpose

The purpose of this Appendix is to provide a list of the Data Deliverables associated to this Contract. The Contract Data Requirements List (CDRL) provides details on the delivery frequency, distribution and approval requirements for each data deliverable. The Data Item Descriptions (DID) describe the content and format of the data deliverables that the Contractor is required to prepare and deliver under the terms and conditions of this Contract.

1.2Data Item Delivery

1.2.1 Delivery Instructions: Every data deliverable must be submitted electronically in its native format (MS-Word, MS-Excel, etc.) to the Office of Primary Interest indicated at Section 5 of the associated DID filled-in form. The relevant electronic mail addresses will be provided at the time the Contract is awarded.

1.2.2 Acknowledgement: The Crown will acknowledge receipt of all data deliverables compliant with the required format within five (5) working days. Non-compliant deliverables will be returned to the Contractor for correction and resubmission.

1.2.3 Approval Requirements: When approval is necessary, a duly authorized and signed covering letter must be faxed or sent through regular mail. This covering letter must contain the exact reference to: (1) date/time of the e-mail, (2) e-mail recipients, (3) Contract Number, (4) CDRL/DID Number, and (5) Title of the document/report.

1.2.4 Crown Review. When approval of a data deliverable is required, the Technical Authority shall provide approval or rejection within:

a. Twenty (20) working days from the receipt of the deliverable; or
b. As mutually agreed.

1.2.5 Rejection: Upon rejection of any data deliverable, the Crown shall provide rationale for rejection and shall provide requirements for the subsequent submission. The Contractor is allowed ten (10) working days from the receipt of the rejection to make corrections and resubmit the data deliverable. After receiving the updated data deliverable, the Crown shall review and provide approval or rejection within ten (10) working days. If at the end of two (2) rework cycles, the data deliverable is still not considered acceptable, the issue shall be referred to the Contracting Authority.

1.2.6 Failure to Respond: Failure by the Crown to respond within the specified time limits shall result in the release of the milestone payment associated with that data deliverable; however, it will not release the Contractor from its obligation to complete the work satisfactorily

Contract Data Requirements List (CDRL)

Each row from the CDRL table presented below refers to the Data Item Description (DID) forms provided below.

The mapping between CDRL items and DID is made using the following convention:

- 1nnn PM-nnn Project Management
- 2nnn ENG-nnn Engineering Support

CDRL Item	DID	Title	First Submission	Frequency	Copies	Approval Level
1001	PM-001	Implementation Management Plan	As required	Update as required	3	Approve
2001	ENG-001	Configuration Management Plan (CMP)	As required	Update as required	3	Approve
2002	ENG-002	Software Development Plan (SDP)	As required	Update as required	3	Approve
2003	ENG-003	System/Subsystem Specification (SSS)	As required	Update as required	3	Approve
2004	ENG-004	System/Subsystem Design Description (SSDD)	As required	Update as required	3	Approve
2005	ENG-005	Software Requirements Specification (SRS)	As required	Update as required	3	Approve
2006	ENG-006	Interface Requirements Specification (IRS)	As required	Update as required	3	Approve
2007	ENG-007	Software Design Description (SDD)	As required	Update as required	3	Approve
2008	ENG-008	Interface Design Description (IDD)	As required	Update as required	3	Approve
2009	ENG-009	Database Design Description (DBDD)	As required	Update as required	3	Approve
2010	ENG-010	Software Test Plan (STP)	As required	Update as required	3	Approve

CDRL Item	DID	Title	First Submission	Frequency	Copies	Approval Level
1001	PM-001	Implementation Management Plan	As required	Update as required	3	Approve
2011	ENG-011	Software Test Description (STD)	As required	Update as required	3	Approve
2012	ENG-012	Software Test Report (STR)	As required	Update as required	3	Approve
2013	ENG-013	Software User Manual (SUM)	As required	Update as required	3	Approve
2014	ENG-014	Computer Programming Manual (CPM)	As required	Update as required	3	Approve

Data Item Description
1. Title Implementation Management Plan *2. Identification numbers* PM-001
3. Description/Purpose The purpose of this document, or set of documentation, is to establish a baseline and timeline covering the implementation of the Contractor's Implementation Management Plan.
4. Issue Date 1 month after contract award (MAC) Update as required *5. Office of Primary Interest* Technical Authority, copy to Contract Authority *6. GIDEP Applicable*
7. Applicable References & Interrelationship CDRL Item 1001 Annex A, Statement of Work, Section 7.0
8. Originator Contractor *9. Applicable forms*
10. Preparation instructions The Implementation Management Plan must be written in Contractor's format. The Implementation Management Plan must cover, but is not limited to, the following topics: a. Scope Definition. The Scope Definition involves subdividing the major project deliverables into smaller, more manageable components. The Scope Definition must cover all the deliverables and produce a Work Breakdown Structure (WBS) based on the Contractor's proposal. The Contractor's proposal must address and cover all of the requirements expressed within Annex A, Statement of Work. b. Activity Definition. The Activity Definition involves identifying and documenting the specific activities that must be performed to produce the deliverables and sub-deliverables identified in the Work Breakdown Structure (WBS). Project network diagrams must display the project's activities and the logical relationships or dependencies among them. c. Resource Requirements. The Resource Planning involves determining what physical resources (people, equipment, facilities) should be used and when they would be needed to perform the project activities. The Resource Requirements should be a description of what types of resources are required and in what quantities for each element at the lowest level of the WBS. d. Organization Plan. The Organization Plan must describe when human resources must be brought onto and taken off of the project team. An organization chart must display the reporting relationships and the relationships to the implementation Project Manager. e. Quality Management Plan. The Quality Management Plan must describe how the Contractor will implement its quality policy as it relates to this implementation. It must describe the Project quality system, i.e. the organizational structure, responsibilities, procedures, processes, and resources needed to implement quality management. f. Communication Plan. The communication plan must describe how Canada will be kept informed of the implementation progress. The methods and procedures must cover the frequency and the distribution requirements expressed within this SOW.

g. Risk Management Plan. The risk management plan must describe how risk identification, analysis, response planning, monitoring and control will be structured and performed during the project implementation phase.
h. Project Schedule. The project schedule must include at least planned start, duration, effort and expected finish dates for each activity. Although the schedule might be developed and maintained using any of the Contractor selected software, the project schedule must be delivered in MS-Project format. Despite the convenience of presenting the project schedule in a summary form, all the scheduling details must be accessible by using the MS-Project software.
i. Project Cost Breakdown. The project cost breakdown shall be the baseline of all the costs associated with the project schedule, in line with the resource requirements to be identified by the Contractor.
j. Change Control. The Contractor must describe the process for keeping track of changes to the Implementation Management Plan.

Traceability. The Implementation Management Plan must be clearly traceable to the requirements expressed within Annex A, Statement of Work, and at the same time, must be clearly traceable to the Contractor's Implementation Management Plan.

Comprehensive Coverage. The Implementation Management Plan must cover all the deliverables that are spelled out within Annex A, Statement of Work.

Roles and Responsibilities. The Implementation Management Plan must clearly identify the Contractor roles and responsibilities in establishing the required services.

Data Item Description
1. Title Configuration Management Plan *2. Identification numbers* ENG-001
3. Description/Purpose The Configuration Management Plan must address the application of configuration management principles contained in the ANSI/EIA 649 standard.
4. Issue Date 3 MAC Update as required *5. Office of Primary Interest* Technical Authority, copy to Contract Authority *6. GIDEP Applicable*
7. Applicable References & Interrelationship CDRL Item 2001 Annex A, Statement of Work, Section 7.0
8. Originator Contractor *9. Applicable forms*
10. Preparation instructions The Contractor must prepare and deliver a Configuration Management Plan (CMP) that addresses the application of configuration management principles as delineated in ANSI/ EIA 649. The Configuration Management Plan must be produced in Contractor's format and must address, but is not limited to, the following areas: a. Configuration Management Summary, including general product definition and scope; b. Contractor Configuration Management Organization, including roles, responsibilities and resources; c. Configuration Identification; d. Configuration Control; e. Configuration Status Accounting; f. Configuration Audits: The Configuration Audits section must include details on the planning and conduct of functional configuration audits (FCA), physical configuration audits (PCA) and post-implementation review audits (PIRA); and g. Subcontractors and vendors: the Contractor must describe proposed systems for control over subcontractors and vendors to support the CM requirements.

Data Item Description
1. Title Software Development Plan *2. Identification numbers* ENG-002
3. Description/Purpose The purpose of this document, or set of documentation, is to define the contractors plan for developing software tools and applications for the R4 HR application.
4. Issue Date 1 MAC Update as required *5. Office of Primary Interest* Technical Authority, copy to Contract Authority *6. GIDEP Applicable*
7. Applicable References & Interrelationship CDRL Item 2002 Annex A, Statement of Work, Section 7.0
8. Originator Contractor *9. Applicable forms*
10. Preparation instructions The Software Development Plan must be written in Contractor's format. The Software Development Plan must cover, but is not limited to, the following topics: ***1. Scope*** This section must be divided into the following paragraphs. 1.1 Identification. This paragraph must contain a full identification of the system and the software to which this document applies, including, as applicable, identification number(s), title(s), abbreviation(s), version number(s), and release number(s). 1.2 System overview. This paragraph must briefly state the purpose of the system and the software to which this document applies. It must describe the general nature of the system and software; summarize the history of system development, operation, and maintenance; identify the project sponsor, acquirer, user, developer, and support agencies; identify current and planned operating sites; and list other relevant documents. 1.3 Document overview. This paragraph must summarize the purpose and contents of this document and must describe any security or privacy considerations associated with its use. 1.4 Relationship to other plans. This paragraph must describe the relationship, if any, of the Software Development Plan (SDP) to other project plans.

2. Referenced documents.

This section must list the number, title, revision, and date of all documents referenced in this plan. This section must also identify the source for all documents not available through normal Government stocking activities.

3. Overview of required work.

This section must be divided into paragraphs as needed to establish the context for the planning described in later sections. It must include, as applicable, an overview of:
a. Requirements and constraints on the system and software to be developed
b. Requirements and constraints on project documentation
c. Position of the project in the system life cycle
d. The selected program/acquisition strategy or any requirements or constraints on it
e. Requirements and constraints on project schedules and resources
f. Other requirements and constraints, such as on project security, privacy, methods, standards, interdependencies in hardware and software development, etc.

4. Plans for performing general software development activities.

This section must be divided into the following paragraphs. Provisions corresponding to non-required activities may be satisfied by the words "Not applicable." If different builds or different software on the project require different planning, these differences must be noted in the paragraphs. In addition to the content specified below, each paragraph must identify applicable risks/uncertainties and plans for dealing with them.

4.1 Software development process.

This paragraph must describe the software development process to be used. The planning must cover all contractual clauses concerning this topic, identifying planned builds, if applicable, their objectives, and the software development activities to be performed in each build.

4.2 General plans for software development.

This paragraph must be divided into the following subparagraphs.

4.2.1 Software development methods.

This paragraph must describe or reference the software development methods to be used. Included must be descriptions of the manual and automated tools and procedures to be used in support of these methods. The methods must cover all contractual clauses concerning this topic. Reference may be made to other paragraphs in this plan if the methods are better described in context with the activities to which they will be applied.

4.2.2 Standards for software products.

This paragraph must describe or reference the standards to be followed for representing requirements, design, code, test cases, test procedures, and test results. The standards must cover all contractual clauses concerning this topic. Reference may be made to other paragraphs in this plan if the standards are better described in context with the activities to which they will be applied. Standards for code must be provided for each programming language to be used. They must include at a minimum:
a. Standards for format (such as indentation, spacing, capitalization, and order of information)
b. Standards for header comments (requiring, for example, name/identifier of the code; version identification; modification history; purpose; requirements and design decisions implemented; notes on the processing (such as algorithms used, assumptions, constraints, limitations, and side effects); and notes on the data (inputs, outputs, variables, data structures, etc.)
c. Standards for other comments (such as required number and content expectations)
d. Naming conventions for variables, parameters, packages, procedures, files, etc.
e. Restrictions, if any, on the use of programming language constructs or features
f. Restrictions, if any, on the complexity of code aggregates

4.2.3 Reusable software products.

This paragraph must be divided into the following subparagraphs.

4.2.3.1 Incorporating reusable software products.

This paragraph must describe the approach to be followed for identifying, evaluating, and incorporating reusable software products, including the scope of the search for such products and the criteria to be used for their evaluation. It must cover all contractual clauses concerning this topic. Candidate or selected reusable software products known at the time this plan is prepared or updated must be identified and described, together with benefits, drawbacks, and restrictions, as applicable, associated with their use.

4.2.3.2 Developing reusable software products.

This paragraph must describe the approach to be followed for identifying, evaluating, and reporting opportunities for developing reusable software products. It must cover all contractual clauses concerning this topic.

4.2.4 Handling of critical requirements.

This paragraph must be divided into the following subparagraphs to describe the approach to be followed for handling requirements designated critical. The planning in each subparagraph must cover all contractual clauses concerning the identified topic.

4.2.4.1 Safety assurance

4.2.4.2 Security assurance

4.2.4.3 Privacy assurance

4.2.4.4 Assurance of other critical requirements

4.2.5 Computer hardware resource utilization.

This paragraph must describe the approach to be followed for allocating computer hardware resources and monitoring their utilization. It must cover all contractual clauses concerning this topic.

4.2.6 Recording rationale.

This paragraph must describe the approach to be followed for recording rationale that will be useful to the support agency for key decisions made on the project. It must interpret the term "key decisions" for the project and state where the rationales are to be recorded. It must cover all contractual clauses concerning this topic.

4.2.7 Access for acquirer review.

This paragraph must describe the approach to be followed for providing the acquirer or its authorized representative access to developer and subcontractor facilities for review of software products and activities. It must cover all contractual clauses concerning this topic.

5. Plans for performing detailed software development activities.

This section must be divided into the following paragraphs. Provisions corresponding to non-required activities may be satisfied by the words "Not applicable." If different builds or different software on the project require different planning, these differences must be noted in the paragraphs. The discussion of each activity must include the approach (methods/procedures/tools) to be applied to:
1) the analysis or other technical tasks involved,
2) the recording of results, and

3) the preparation of associated deliverables, if applicable. The discussion must also identify applicable risks/uncertainties and plans for dealing with them. Reference may be made to 4.2.1 if applicable methods are described there.

5.1 Project planning and oversight.

This paragraph must be divided into the following subparagraphs to describe the approach to be followed for project planning and oversight. The planning in each subparagraph must cover all contractual clauses regarding the identified topic.

5.1.1 Software development planning (covering updates to this plan)

5.1.2 Computer Software Configuration Item (CSCI) test planning

5.1.3 System test planning

5.1.4 Software installation planning

5.1.5 Software transition planning

5.1.6 Following and updating plans, including the intervals for management review

5.2 Establishing a software development environment.

This paragraph must be divided into the following subparagraphs to describe the approach to be followed for establishing, controlling, and maintaining a software development environment. The planning in each subparagraph must cover all contractual clauses regarding the identified topic.

5.2.1 Software engineering environment

5.2.2 Software test environment

5.2.3 Software development library

5.2.4 Software development files

5.2.5 Non-deliverable software

5.3 System requirements analysis.

This paragraph must be divided into the following subparagraphs to describe the approach to be followed for participating in system requirements analysis. The planning in each subparagraph must cover all contractual clauses regarding the identified topic.

5.3.1 Analysis of user input

5.3.2 Operational concept

5.3.3 System requirements

5.4 System design.

This paragraph must be divided into the following subparagraphs to describe the approach to be followed for participating in system design. The planning in each subparagraph must cover all contractual clauses regarding the identified topic.

5.4.1 System-wide design decisions

5.4.2 System architectural design

5.5 Software requirements analysis.

This paragraph must describe the approach to be followed for software requirements analysis. The approach must cover all contractual clauses concerning this topic.

5.6 Software design.

This paragraph must be divided into the following subparagraphs to describe the approach to be followed for software design. The planning in each subparagraph must cover all contractual clauses regarding the identified topic.

5.6.1 Computer Software Configuration Item (CSCI)-wide design decisions

5.6.2 CSCI architectural design

5.6.3 CSCI detailed design

5.7 Software implementation and unit testing.

This paragraph must be divided into the following subparagraphs to describe the approach to be followed for software implementation and unit testing. The planning in each subparagraph must cover all contractual clauses regarding the identified topic.

5.7.1 Software implementation

5.7.2 Preparing for unit testing

5.7.3 Performing unit testing

5.7.4 Revision and retesting

5.7.5 Analyzing and recording unit test results

5.8 Unit integration and testing.

This paragraph must be divided into the following subparagraphs to describe the approach to be followed for unit integration and testing. The planning in each subparagraph must cover all contractual clauses regarding the identified topic.

5.8.1 Preparing for unit integration and testing

5.8.2 Performing unit integration and testing

5.8.3 Revision and retesting

5.8.4 Analyzing and recording unit integration and test results

5.9 CSCI qualification testing.

This paragraph must be divided into the following subparagraphs to describe the approach to be followed for CSCI qualification testing. The planning in each subparagraph must cover all contractual clauses regarding the identified topic.

5.9.1 Independence in CSCI qualification testing

5.9.2 Testing on the target computer system

5.9.3 Preparing for CSCI qualification testing

5.9.4 Dry run of CSCI qualification testing

5.9.5 Performing CSCI qualification testing

5.9.6 Revision and retesting

5.9.7 Analyzing and recording CSCI qualification test results

5.10 Computer Software Configuration Item (CSCI) / Hardware Configuration Item (HWCI) integration and testing.

This paragraph must be divided into the following subparagraphs to describe the approach to be followed for participating in CSCI/HWCI integration and testing. The planning in each subparagraph must cover all contractual clauses regarding the identified topic.

5.10.1 Preparing for CSCI/HWCI integration and testing

5.10.2 Performing CSCI/HWCI integration and testing

5.10.3 Revision and retesting

5.10.4 Analyzing and recording CSCI/HWCI integration and test results

5.11 System qualification testing.

This paragraph must be divided into the following subparagraphs to describe the approach to be followed for participating in system qualification testing. The planning in each subparagraph must cover all contractual clauses regarding the identified topic.

5.11.1 Independence in system qualification testing

5.11.2 Testing on the target computer system

5.11.3 Preparing for system qualification testing

5.11.4 Dry run of system qualification testing

5.11.5 Performing system qualification testing

5.11.6 Revision and retesting

5.11.7 Analyzing and recording system qualification test results

5.12 Preparing for software use.

This paragraph must be divided into the following subparagraphs to describe the approach to be followed for preparing for software use. The planning in each subparagraph must cover all contractual clauses regarding the identified topic.

5.12.1 Preparing the executable software

5.12.2 Preparing version descriptions for user sites

5.12.3 Preparing user manuals

5.12.4 Installation at user sites

5.13 Preparing for software transition.

This paragraph must be divided into the following subparagraphs to describe the approach to be followed for preparing for software transition. The planning in each subparagraph must cover all contractual clauses regarding the identified topic.

5.13.1 Preparing the executable software

5.13.2 Preparing source files

5.13.3 Preparing version descriptions for the support site

5.13.4 Preparing the "as built" CSCI design and other software support information

5.13.5 Updating the system design description

5.13.6 Preparing support manuals

5.13.7 Transition to the designated support site

5.14 Software configuration management.

This paragraph must be divided into the following subparagraphs to describe the approach to be followed for software configuration management. The planning in each subparagraph must cover all contractual clauses regarding the identified topic.

5.14.1 Configuration identification

5.14.2 Configuration control

5.14.3 Configuration status accounting

5.14.4 Configuration audits

5.14.5 Packaging, storage, handling, and delivery

5.15 Software product evaluation.

This paragraph must be divided into the following subparagraphs to describe the approach to be followed for software product evaluation. The planning in each subparagraph must cover all contractual clauses regarding the identified topic.

5.15.1 In-process and final software product evaluations

5.15.2 Software product evaluation records, including items to be recorded

5.15.3 Independence in software product evaluation

5.16 Software quality assurance.

This paragraph must be divided into the following subparagraphs to describe the approach to be followed for software quality assurance. The planning in each subparagraph must cover all contractual clauses regarding the identified topic.

5.16.1 Software quality assurance evaluations

5.16.2 Software quality assurance records, including items to be recorded

5.16.3 Independence in software quality assurance

5.17 Corrective action.

This paragraph must be divided into the following subparagraphs to describe the approach to be followed for corrective action. The planning in each subparagraph must cover all contractual clauses regarding the identified topic.

5.17.1 Problem/change reports

This paragraph includes items to be recorded (candidate items include project name, originator, problem number, problem name, software element or document affected, origination date, category and priority, description, analyst assigned to the problem, date assigned, date completed, analysis time, recommended solution, impacts, problem status, approval of solution, follow-up actions, corrector, correction date, version where corrected, correction time, description of solution implemented)

5.17.2 Corrective action system

5.18 Joint technical and management reviews.

This paragraph must be divided into the following subparagraphs to describe the approach to be followed for joint technical and management reviews. The planning in each subparagraph must cover all contractual clauses regarding the identified topic.

5.18.1 Joint technical reviews, including a proposed set of reviews

5.18.2 Joint management reviews, including a proposed set of reviews

5.19 Other software development activities.

This paragraph must be divided into the following subparagraphs to describe the approach to be followed for other software development activities. The planning in each subparagraph must cover all contractual clauses regarding the identified topic.

5.19.1 Risk management, including known risks and corresponding strategies

5.19.2 Software management indicators, including indicators to be used

5.19.3 Security and privacy

5.19.4 Subcontractor management

5.19.5 Interface with software independent verification and validation (IV&V) agents

5.19.6 Coordination with associate developers

5.19.7 Improvement of project processes

5.19.8 Other activities not covered elsewhere in the plan

6. Schedules and activity network.

This section must present:
a. Schedule(s) identifying the activities in each build and showing initiation of each activity, availability of draft and final deliverables and other milestones, and completion of each activity
b. An activity network, depicting sequential relationships and dependencies among activities and identifying those activities that impose the greatest time restrictions on the project

7. Project organization and resources.

This section must be divided into the following paragraphs to describe the project organization and resources to be applied in each build.

7.1 Project organization.

This paragraph must describe the organizational structure to be used on the project, including the organizations involved, their relationships to one another, and the authority and responsibility of each organization for carrying out required activities.

7.2 Project resources.

This paragraph must describe the resources to be applied to the project. It must include, as applicable:
a. Personnel resources, including:

1) The estimated staff-loading for the project (number of personnel over time);

2) The breakdown of the staff-loading numbers by responsibility (for example, management, software engineering, software testing, software configuration management, software product evaluation, software quality assurance); and

3) A breakdown of the skill levels, geographic locations, and security clearances of personnel performing each responsibility

b. Overview of developer facilities to be used, including geographic locations in which the work will be performed, facilities to be used, and secure areas and other features of the facilities as applicable to the contracted effort.
c. Acquirer-furnished equipment, software, services, documentation, data, and facilities required for the contracted effort. A schedule detailing when these items will be needed must also be included.
d. Other required resources, including a plan for obtaining the resources, dates needed, and availability of each resource item.

8. Notes.

This section must contain any general information that aids in understanding this document (e.g., background information, glossary, rationale). This section must include an alphabetical listing of all acronyms, abbreviations, and their meanings as used in this document and a list of any terms and definitions needed to understand this document.

A. Appendixes.

Appendixes may be used to provide information published separately for convenience in document maintenance (e.g., charts, classified data). As applicable, each appendix must be referenced in the main body of the document where the data would normally have been provided. Appendixes may be bound as separate documents for ease in handling. Appendixes must be lettered alphabetically (A, B, etc.).

Data Item Description
1. Title System/Subsystem Specification *2. Identification numbers* ENG-003
3. Description/Purpose The purpose of this document, or set of documentation, is to define the System/Subsystem Specification for the R4 HR application.
4. Issue Date As Required Update as required *5. Office of Primary Interest* Technical Authority, copy to Contract Authority *6. GIDEP Applicable*
7. Applicable References & Interrelationship CDRL Item 2003 Annex A, Statement of Work, Section 7.0
8. Originator Contractor *9. Applicable forms*
10. Preparation instructions The System/Subsystem Specification must be written in Contractor's format. The System/Subsystem Specification must cover, but is not limited to, the following topics: ***1. Scope.*** This section must be divided into the following paragraphs. 1.1 Identification. This paragraph must contain a full identification of the system to which this document applies, including, as applicable, identification number(s), title(s), abbreviation(s), version number(s), and release number(s). 1.2 System overview. This paragraph must briefly state the purpose of the system to which this document applies. It must describe the general nature of the system; summarize the history of system development, operation, and maintenance; identify the project sponsor, acquirer, user, developer, and support agencies; identify current and planned operating sites; and list other relevant documents. 1.3 Document overview. This paragraph must summarize the purpose and contents of this document and must describe any security or privacy considerations associated with its use.

2. Referenced documents.

This section must list the number, title, revision, and date of all documents referenced in this specification. This section must also identify the source for all documents not available through normal Government stocking activities.

3. Requirements.

This section must be divided into the following paragraphs to specify the system requirements, that is, those characteristics of the system that are conditions for its acceptance. Each requirement must be assigned a project-unique identifier to support testing and traceability and must be stated in such a way that an objective test can be defined for it. Each requirement must be annotated with associated qualification method(s) (see section 4) and, for subsystems, traceability to system requirements (see section 5.a), if not provided in those sections. The degree of detail to be provided must be guided by the following rule: Include those characteristics of the system that are conditions for system acceptance; defer to design descriptions those characteristics that the acquirer is willing to leave up to the developer. If there are no requirements in a given paragraph, the paragraph must so state. If a given requirement fits into more than one paragraph, it may be stated once and referenced from the other paragraphs.

3.1 Required states and modes.

If the system is required to operate in more than one state or mode having requirements distinct from other states or modes, this paragraph must identify and define each state and mode. Examples of states and modes include: idle, ready, active, post-use analysis, training, degraded, emergency, backup, wartime, peacetime. The distinction between states and modes is arbitrary. A system may be described in terms of states only, modes only, states within modes, modes within states, or any other scheme that is useful. If no states or modes are required, this paragraph must so state, without the need to create artificial distinctions. If states and/or modes are required, each requirement or group of requirements in this specification must be correlated to the states and modes. The correlation may be indicated by a table or other method in this paragraph, in an appendix referenced from this paragraph, or by annotation of the requirements in the paragraphs where they appear.

3.2 System capability requirements.

This paragraph must be divided into subparagraphs to itemize the requirements associated with each capability of the system. A "capability" is defined as a group of related requirements. The word "capability" may be replaced with "function," "subject," "object," or other term useful for presenting the requirements.

3.2.x (System capability).

This paragraph must identify a required system capability and must itemize the requirements associated with the capability. If the capability can be more clearly specified by dividing it into constituent capabilities, the constituent capabilities must be specified in subparagraphs. The requirements must specify required behavior of the system and must include applicable parameters, such as response times, throughput times, other timing constraints, sequencing, accuracy, capacities (how much/how many), priorities, continuous operation requirements, and allowable deviations based on operating conditions. The requirements must include, as applicable, required behavior under unexpected, unallowed, or "out of bounds" conditions, requirements for error handling, and any provisions to be incorporated into the system to provide continuity of operations in the event of emergencies. Paragraph 3.3.x of this DID provides a list of topics to be considered when specifying requirements regarding inputs the system must accept and outputs it must produce.

3.3 System external interface requirements.

This paragraph must be divided into subparagraphs to specify the requirements, if any, for the system's external interfaces. This paragraph may reference one or more Interface Requirements Specifications (IRSs) or other documents containing these requirements.

3.3.1 Interface identification and diagrams.

This paragraph must identify the required external interfaces of the system. The identification of each interface must include a project-unique identifier and must designate the interfacing entities (systems, configuration items, users, etc.) by name, number, version, and documentation references, as applicable. The identification must state which entities have fixed interface characteristics (and therefore impose interface requirements on interfacing entities) and which are being developed or modified (thus having interface requirements imposed on them). One or more interface diagrams must be provided to depict the interfaces.

3.3.x (Project-unique identifier of interface).

This paragraph (beginning with 3.3.2) must identify a system external interface by project-unique identifier, must briefly identify the interfacing entities, and must be divided into subparagraphs as needed to state the requirements imposed on the system to achieve the interface. Interface characteristics of the other entities involved in the interface must be stated as assumptions or as "When [the entity not covered] does this, the system shall...," not as requirements on the other entities. This paragraph may reference other documents (such as data dictionaries, standards for communication protocols, and standards for user interfaces) in place of stating the information here. The requirements must include the following, as applicable, presented in any order suited to the requirements, and must note any differences in these characteristics from the point of view of the interfacing entities (such as different expectations about the size, frequency, or other characteristics of data elements):
a. Priority that the system must assign the interface;
b. Requirements on the type of interface (such as real-time data transfer, storage-and-retrieval of data, etc.) to be implemented; and
c. Required characteristics of individual data elements that the system must provide, store, send, access, receive, etc., such as:

1) Names/identifiers;

a) Project-unique identifier

b) Non-technical (natural-language) name

c) DoD standard data element name

d) Technical name (e.g., variable or field name in code or database)

e) Abbreviation or synonymous names

2) Data type (alphanumeric, integer, etc.);

3) Size and format (such as length and punctuation of a character string);

4) Units of measurement (such as meters, dollars, nanoseconds);

5) Range or enumeration of possible values (such as 0-99);

6) Accuracy (how correct) and precision (number of significant digits);

7) Priority, timing, frequency, volume, sequencing, and other constraints, such as whether the data element may be updated and whether business rules apply;

8) Security and privacy constraints; and

9) Sources (setting/sending entities) and recipients (using/receiving entities).

d. Required characteristics of data element assemblies (records, messages, files, arrays, displays, reports, etc.) that the system must provide, store, send, access, receive, etc., such as:

1) Names/identifiers;

a) Project-unique identifier;

b) Non-technical (natural language) name;

c) Technical name (e.g., record or data structure name in code or database); and

d) Abbreviations or synonymous names.

2) Data elements in the assembly and their structure (number, order, grouping)

3) Medium (such as disk) and structure of data elements/assemblies on the medium

4) Visual and auditory characteristics of displays and other outputs (such as colors, layouts, fonts, icons and other display elements, beeps, lights)

5) Relationships among assemblies, such as sorting/access characteristics

6) Priority, timing, frequency, volume, sequencing, and other constraints, such as whether the assembly may be updated and whether business rules apply

7) Security and privacy constraints

8) Sources (setting/sending entities) and recipients (using/receiving entities)

e. Required characteristics of communication methods that the system must use for the interface, such as:

1) Project-unique identifier(s);

2) Communication links/bands/frequencies/media and their characteristics;

3) Message formatting;

4) Flow control (such as sequence numbering and buffer allocation)

5) Data transfer rate, whether periodic/aperiodic, and interval between transfers;

6) Routing, addressing, and naming conventions;

7) Transmission services, including priority and grade; and

8) Safety/security/privacy considerations, such as encryption, user authentication, compartmentalization, and auditing.

f. Required characteristics of protocols the system must use for the interface, such as:

1) Project-unique identifier(s);

2) Priority/layer of the protocol;

3) Packeting, including fragmentation and reassembly, routing, and addressing;

4) Legality checks, error control, and recovery procedures;

5) Synchronization, including connection establishment, maintenance, termination; and

6) Status, identification, and any other reporting features.

g. Other required characteristics, such as physical compatibility of the interfacing entities (dimensions, tolerances, loads, plug compatibility, etc.), voltages, etc.

3.4 System internal interface requirements.

This paragraph must specify the requirements, if any, imposed on interfaces internal to the system. If all internal interfaces are left to the design or to requirement specifications for system components, this fact must be so stated. If such requirements are to be imposed, paragraph 3.3 of this DID provides a list of topics to be considered.

3.5 System internal data requirements.

This paragraph must specify the requirements, if any, imposed on data internal to the system. Included must be requirements, if any, on databases and data files to be included in the system. If all decisions about internal data are left to the design or to requirements specifications for system components, this fact must be so stated. If such requirements are to be imposed, paragraphs 3.3.x.c and 3.3.x.d of this DID provide a list of topics to be considered.

3.6 Adaptation requirements.

This paragraph must specify the requirements, if any, concerning installation-dependent data that the system is required to provide (such as site-dependent latitude and longitude or site-dependent state tax codes) and operational parameters that the system is required to use that may vary according to operational needs (such as parameters indicating operation-dependent targeting constants or data recording).

3.7 Security and privacy requirements.

This paragraph must specify the system requirements, if any, concerned with maintaining security and privacy. The requirements must include, as applicable, the security/privacy environment in which the system must operate, the type and degree of security or privacy to be provided, the security/privacy risks the system must withstand, required safeguards to reduce those risks, the security/privacy policy that must be met, the security/privacy accountability the system must provide, and the criteria that must be met for security/privacy certification/accreditation.

3.8 System environment requirements.

This paragraph must specify the requirements, if any, regarding the environment in which the system must operate. Examples for a software system are the computer hardware and operating system on which the software must run. (Additional requirements concerning computer resources are given in the next paragraph). Examples for a hardware-software system include the environmental conditions that the system must withstand during transportation, storage, and operation, such as conditions in the natural environment (wind, rain, temperature, geographic location), the induced environment (motion, shock, noise, electromagnetic radiation), and environments due to enemy action (explosions, radiation).

3.9 Computer resource requirements.

This paragraph must be divided into the following subparagraphs. Depending upon the nature of the system, the computer resources covered in these subparagraphs may constitute the environment of the system (as for a software system) or components of the system (as for a hardware-software system).

3.9.1 Computer hardware requirements.

This paragraph must specify the requirements, if any, regarding computer hardware that must be used by, or incorporated into, the system. The requirements must include, as applicable, number of each type of equipment, type, size, capacity, and other required characteristics of processors, memory, input/output devices, auxiliary storage, communications/network equipment, and other required equipment.

3.9.2 Computer hardware resource utilization requirements.

This paragraph must specify the requirements, if any, on the system's computer hardware resource utilization, such as maximum allowable use of processor capacity, memory capacity, input/output device

capacity, auxiliary storage device capacity, and communications/network equipment capacity. The requirements (stated, for example, as percentages of the capacity of each computer hardware resource) must include the conditions, if any, under which the resource utilization is to be measured.

3.9.3 Computer software requirements.

This paragraph must specify the requirements, if any, regarding computer software that must be used by, or incorporated into, the system. Examples include operating systems, database management systems, communications/ network software, utility software, input and equipment simulators, test software, and manufacturing software. The correct nomenclature, version, and documentation references of each such software item must be provided.

3.9.4 Computer communications requirements.

This paragraph must specify the additional requirements, if any, concerning the computer communications that must be used by, or incorporated into, the system. Examples include geographic locations to be linked; configuration and network topology; transmission techniques; data transfer rates; gateways; required system use times; type and volume of data to be transmitted/received; time boundaries for transmission/reception/response; peak volumes of data; and diagnostic features.

3.10 System quality factors.

This paragraph must specify the requirements, if any, pertaining to system quality factors. Examples include quantitative requirements concerning system functionality (the ability to perform all required functions), reliability (the ability to perform with correct, consistent results -- such as mean time between failure for equipment), maintainability (the ability to be easily serviced, repaired, or corrected), availability (the ability to be accessed and operated when needed), flexibility (the ability to be easily adapted to changing requirements), portability of software (the ability to be easily modified for a new environment), reusability (the ability to be used in multiple applications), testability (the ability to be easily and thoroughly tested), usability (the ability to be easily learned and used), and other attributes.

3.11 Design and construction constraints.

This paragraph must specify the requirements, if any, that constrain the design and construction of the system. For hardware-software systems, this paragraph must include the physical requirements imposed on the system. These requirements may be specified by reference to appropriate commercial or military standards and specifications. Examples include requirements concerning:
a. Use of a particular system architecture or requirements on the architecture, such as required subsystems; use of standard, military, or existing components; or use of Government/acquirer-furnished property (equipment, information, or software)
b. Use of particular design or construction standards; use of particular data standards; use of a particular programming language; workmanship requirements and production techniques
c. Physical characteristics of the system (such as weight limits, dimensional limits, color, protective coatings); interchangeability of parts; ability to be transported from one location to another; ability to be carried or set up by one, or a given number of, persons
d. Materials that can and cannot be used; requirements on the handling of toxic materials; limits on the electromagnetic radiation that the system is permitted to generate
e. Use of nameplates, part marking, serial and lot number marking, and other identifying markings
f. Flexibility and expandability that must be provided to support anticipated areas of growth or changes in technology, threat, or mission

3.12 Training-related requirements.

This paragraph must specify the system requirements, if any, pertaining to training. Examples include training devices and training materials to be included in the system.

3.13 Logistics-related requirements.

This paragraph must specify the system requirements, if any, concerned with logistics considerations. These considerations may include: system maintenance, software support, system transportation modes, supply system requirements, impact on existing facilities, and impact on existing equipment.

3.14 Other requirements.

This paragraph must specify additional system requirements, if any, not covered in the previous paragraphs. Examples include requirements for system documentation, such as specifications, drawings, technical manuals, test plans and procedures, and installation instruction data, if not covered in other contractual documents.

3.15 Packaging requirements.

This section must specify the requirements, if any, for packaging, labeling, and handling the system and its components for delivery. Applicable military specifications and standards may be referenced if appropriate.

3.16 Precedence and criticality of requirements.

This paragraph must specify, if applicable, the order of precedence, criticality, or assigned weights indicating the relative importance of the requirements in this specification. Examples include identifying those requirements deemed critical to safety, to security, or to privacy for purposes of singling them out for special treatment. If all requirements have equal weight, this paragraph must so state.

4. Qualification provisions.

This section must define a set of qualification methods and must specify for each requirement in Section 3 the method(s) to be used to ensure that the requirement has been met. A table may be used to present this information, or each requirement in Section 3 may be annotated with the method(s) to be used. Qualification methods may include:
a. Demonstration: The operation of the system, or a part of the system, that relies on observable functional operation not requiring the use of instrumentation, special test equipment, or subsequent analysis.
b. Test: The operation of the system, or a part of the system, using instrumentation or other special test equipment to collect data for later analysis.
c. Analysis: The processing of accumulated data obtained from other qualification methods. Examples are reduction, interpolation, or extrapolation of test results.
d. Inspection: The visual examination of system components, documentation, etc.
e. Special qualification methods. Any special qualification methods for the system, such as special tools, techniques, procedures, facilities, acceptance limits, use of standard samples, preproduction or periodic production samples, pilot models, or pilot lots.

5. Requirements traceability.

For system-level specifications, this paragraph does not apply. For subsystem-level specifications, this paragraph must contain:
a. Traceability from each subsystem requirement in this specification to the system requirements it addresses. (Alternatively, this traceability may be provided by annotating each requirement in Section 3.) Note: Each level of system refinement may result in requirements not directly traceable to higher-level requirements. For example, a system architectural design that creates two subsystems may result in requirements about how the subsystems will interface, even though these interfaces are not covered in system requirements. Such requirements may be traced to a general requirement such as "system implementation" or to the system design decisions that resulted in their generation.
b. Traceability from each system requirement that has been allocated to the subsystem covered by this specification to the subsystem requirements that address it. All system requirements allocated to the subsystem must be accounted for. Those that trace to subsystem requirements contained in IRSs must reference those IRSs.

6. Notes.

This section must contain any general information that aids in understanding this document (e.g., background information, glossary, rationale). This section must contain an alphabetical listing of all acronyms, abbreviations, and their meanings as used in this document and a list of any terms and definitions needed to understand this document.

A. Appendixes.

Appendixes may be used to provide information published separately for convenience in document maintenance (e.g., charts, classified data). As applicable, each appendix must be referenced in the main body of the document where the data would normally have been provided. Appendixes may be bound as separate documents for ease in handling. Appendixes must be lettered alphabetically (A, B, etc.).

Data Item Description

1. Title

System/Subsystem Design Description

2. Identification numbers

ENG-004

3. Description/Purpose

The purpose of this document, or set of documentation, is to define the System/Subsystem Design Description for the R4 HR application.

4. Issue Date

As Required

Update as required

5. Office of Primary Interest

Technical Authority, copy to

Contract Authority

6. GIDEP Applicable

7. Applicable References & Interrelationship

CDRL Item 2004

Annex A, Statement of Work, Section 7.0

8. Originator

Contractor

9. Applicable forms

10. Preparation instructions

The System/Subsystem Design Description must be written in Contractor's format. The System/Subsystem Design Description must cover, but is not limited to, the following topics:

1. Scope.

This section must be divided into the following paragraphs.

1.1 Identification.

This paragraph must contain a full identification of the system to which this document applies, including, as applicable, identification number(s), title(s), abbreviation(s), version number(s), and release number(s).

1.2 System overview.

This paragraph must briefly state the purpose of the system to which this document applies. It must describe the general nature of the system; summarize the history of system development, operation, and maintenance; identify the project sponsor, acquirer, user, developer, and support agencies; identify current and planned operating sites; and list other relevant documents.

1.3 Document overview.

This paragraph must summarize the purpose and contents of this document and must describe any security or privacy considerations associated with its use.

2. Referenced documents.

This section must list the number, title, revision, and date of all documents referenced in this document. This section must also identify the source for all documents not available through normal Government stocking activities.

3. System-wide design decisions.

This section must be divided into paragraphs as needed to present system-wide design decisions, that is, decisions about the system's behavioral design (how it will behave, from a user's point of view, in meeting its requirements, ignoring internal implementation) and other decisions affecting the selection and design of system components. If all such decisions are explicit in the requirements or are deferred to the design of the system components, this section must so state. Design decisions that respond to requirements designated critical, such as those for safety, security, or privacy, must be placed in separate subparagraphs. If a design decision depends upon system states or modes, this dependency must be indicated. Design conventions needed to understand the design must be presented or referenced. Examples of system-wide design decisions are the following:

a. Design decisions regarding inputs the system will accept and outputs it will produce, including interfaces with other systems, configuration items, and users (4.3.x of this DID identifies topics to be considered in this description). If part or all of this information is given in Interface Design Descriptions (IDDs), they may be referenced.

b. Design decisions on system behavior in response to each input or condition, including actions the system will perform, response times and other performance characteristics, description of physical systems modeled, selected equations/algorithms/ rules, and handling of unallowed inputs or conditions.

c. Design decisions on how system databases/data files will appear to the user (4.3.x of this DID identifies topics to be considered in this description). If part or all of this information is given in Database Design Descriptions (DBDDs), they may be referenced.

d. Selected approach to meeting safety, security, and privacy requirements.

e. Design and construction choices for hardware or hardware-software systems, such as physical size, color, shape, weight, materials, and markings.

f. Other system-wide design decisions made in response to requirements, such as selected approach to providing required flexibility, availability, and maintainability.

4. System architectural design.

This section must be divided into the following paragraphs to describe the system architectural design. If part or all of the design depends upon system states or modes, this dependency must be indicated. If design information falls into more than one paragraph, it may be presented once and referenced from the other paragraphs. Design conventions needed to understand the design must be presented or referenced.

Note: For brevity, this section is written in terms of organizing a system directly into Hardware Configuration Items (HWCIs), Computer Software Configuration Items (CSCIs), and manual operations, but should be interpreted to cover organizing a system into subsystems, organizing a subsystem into HWCIs, CSCIs, and manual operations, or other variations as appropriate.

4.1 System components.

This paragraph must:
a. Identify the components of the system (HWCIs, CSCIs, and manual operations). Each component must be assigned a project-unique identifier. Note: a database may be treated as a CSCI or as part of a CSCI.
b. Show the static (such as "consists of") relationship(s) of the components. Multiple relationships may be presented, depending on the selected design methodology.
c. State the purpose of each component and identify the system requirements and system-wide design decisions allocated to it. (Alternatively, the allocation of requirements may be provided in 5.a.)
d. Identify each component's development status/type, if known (such as new development, existing component to be reused as is, existing design to be reused as is, existing design or component to be reengineered, component to be developed for reuse, component planned for Build N, etc.) For existing design or components, the description must provide identifying information, such as name, version, documentation references, location, etc.
e. For each computer system or other aggregate of computer hardware resources identified for use in the system, describe its computer hardware resources (such as processors, memory, input/output devices, auxiliary storage, and communications/ network equipment). Each description must, as applicable, identify the configuration items that will use the resource, describe the allocation of resource utilization to each CSCI that will use the resource (for example, 20% of the resource's capacity allocated to CSCI 1, 30% to CSCI 2), describe the conditions under which utilization will be measured, and describe the characteristics of the resource:

1) Descriptions of computer processors must include, as applicable, manufacturer name and model number, processor speed/capacity, identification of instruction set architecture, applicable compiler(s), word size (number of bits in each computer word), character set standard (such as ASCII, EBCDIC), and interrupt capabilities;

2) Descriptions of memory must include, as applicable, manufacturer name and model number and memory size, type, speed, and configuration (such as 256K cache memory, 16MB RAM (4MB x 4));

3) Descriptions of input/output devices must include, as applicable, manufacturer name and model number, type of device, and device speed/capacity;

4) Descriptions of auxiliary storage must include, as applicable, manufacturer name and model number, type of storage, amount of installed storage, and storage speed;

5) Descriptions of communications/network equipment, such as modems, network interface cards, hubs, gateways, cabling, high speed data lines, or aggregates of these or other components, must include, as applicable, manufacturer name and model number, data transfer rates/capacities, network topologies, transmission techniques, and protocols used; and

6) Each description must also include, as applicable, growth capabilities, diagnostic capabilities, and any additional hardware capabilities relevant to the description.

f. Present a specification tree for the system, that is, a diagram that identifies and shows the relationships among the planned specifications for the system components.

4.2 Concept of execution.

This paragraph must describe the concept of execution among the system components. It must include diagrams and descriptions showing the dynamic relationship of the components, that is, how they will interact during system operation, including, as applicable, flow of execution control, data flow, dynamically controlled sequencing, state transition diagrams, timing diagrams, priorities among components, handling of interrupts, timing/sequencing relationships, exception handling, concurrent execution, dynamic allocation/de-allocation, dynamic creation/deletion of objects, processes, tasks, and other aspects of dynamic behavior.

4.3 Interface design.

This paragraph must be divided into the following subparagraphs to describe the interface characteristics of the system components. It must include both interfaces among the components and their interfaces with external entities such as other systems, configuration items, and users. Note: There is no requirement for these interfaces to be completely designed at this level; this paragraph is provided to allow the recording of interface design decisions made as part of system architectural design. If part or all of this information is contained in Interface Design Descriptions (IDDs) or elsewhere, these sources may be referenced.

4.3.1 Interface identification and diagrams.

This paragraph must state the project-unique identifier assigned to each interface and must identify the interfacing entities (systems, configuration items, users, etc.) by name, number, version, and documentation references, as applicable. The identification must state which entities have fixed interface characteristics (and therefore impose interface requirements on interfacing entities) and which are being developed or modified (thus having interface requirements imposed on them). One or more interface diagrams must be provided, as appropriate, to depict the interfaces.

4.3.x (Project-unique identifier of interface).

This paragraph (beginning with 4.3.2) must identify an interface by project-unique identifier, must briefly identify the interfacing entities, and must be divided into subparagraphs as needed to describe the interface characteristics of one or both of the interfacing entities. If a given interfacing entity is not covered by this SSDD (for example, an external system) but its interface characteristics need to be mentioned to describe interfacing entities that are, these characteristics must be stated as assumptions or as "When [the entity not covered] does this, [the entity that is covered] will ...." This paragraph may reference other documents (such as data dictionaries, standards for protocols, and standards for user interfaces) in place of stating the information here. The design description must include the following, as applicable, presented in any order suited to the information to be provided, and must note any differences in these characteristics from the point of view of the interfacing entities (such as different expectations about the size, frequency, or other characteristics of data elements):
a. Priority assigned to the interface by the interfacing entity(ies)
b. Type of interface (such as real-time data transfer, storage-and-retrieval of data, etc.) to be implemented

c. Characteristics of individual data elements that the interfacing entity(ies) will provide, store, send, access, receive, etc., such as:

1) Names/identifiers

- a) Project-unique identifier
- b) Non-technical (natural-language) name
- c) DoD standard data element name
- d) Technical name (e.g., variable or field name in code or database)
- e) Abbreviation or synonymous names

2) Data type (alphanumeric, integer, etc.);

3) Size and format (such as length and punctuation of a character string);

4) Units of measurement (such as meters, dollars, nanoseconds);

5) Range or enumeration of possible values (such as 0-99);

6) Accuracy (how correct) and precision (number of significant digits);

7) Priority, timing, frequency, volume, sequencing, and other constraints, such as whether the data element may be updated and whether business rules apply;

8) Security and privacy constraints; and

9) Sources (setting/sending entities) and recipients (using/receiving entities)

d. Characteristics of data element assemblies (records, messages, files, arrays, displays, reports, etc.) that the interfacing entity(ies) will provide, store, send, access, receive, etc., such as:

1) Names/identifiers

- a) Project-unique identifier to be used for traceability
- b) Non-technical (natural language) name
- c) Technical name (e.g., record or data structure name in code or database)
- d) Abbreviations or synonymous names;

2) Data elements in the assembly and their structure (number, order, grouping);

3) Medium (such as disk) and structure of data elements/assemblies on the medium;

4) Visual and auditory characteristics of displays and other outputs (such as colors, layouts, fonts, icons and other display elements, beeps, lights);

5) Relationships among assemblies, such as sorting/access characteristics;

6) Priority, timing, frequency, volume, sequencing, and other constraints, such as whether the assembly may be updated and whether business rules apply;

7) Security and privacy constraints; and

8) Sources (setting/sending entities) and recipients (using/receiving entities).

e. Characteristics of communication methods that the interfacing entity(ies) will use for the interface, such as:

1) Project-unique identifier(s);

2) Communication links/bands/frequencies/media and their characteristics ;

3) Message formatting;

4) Flow control (such as sequence numbering and buffer allocation);

5) Data transfer rate, whether periodic/aperiodic, and interval between transfers;

6) Routing, addressing, and naming conventions;

7) Transmission services, including priority and grade; and

8) Safety/security/privacy considerations, such as encryption, user authentication, compartmentalization, and auditing

f. Characteristics of protocols that the interfacing entity(ies) will use for the interface, such as:

1) Project-unique identifier(s);

2) Priority/layer of the protocol;

3) Packeting, including fragmentation and reassembly, routing, and addressing;

4) Legality checks, error control, and recovery procedures;

5) Synchronization, including connection establishment, maintenance, termination; and

6) Status, identification, and any other reporting features.

g. Other characteristics, such as physical compatibility of the interfacing entity (ies) (dimensions, tolerances, loads, voltages, plug compatibility, etc.)

5. Requirements traceability.

This paragraph must contain:
a. Traceability from each system component identified in this SSDD to the system requirements allocated to it. (Alternatively, this traceability may be provided in 4.1.)
b. Traceability from each system requirement to the system components to which it is allocated.

6. Notes.

This section must contain any general information that aids in understanding this document (e.g., background information, glossary, rationale). This section must contain an alphabetical listing of all acronyms, abbreviations, and their meanings as used in this document and a list of any terms and definitions needed to understand this document.

A. Appendixes.

Appendixes may be used to provide information published separately for convenience in document maintenance (e.g., charts, classified data). As applicable, each appendix must be referenced in the main body of the document where the data would normally have been provided. Appendixes may be bound as separate documents for ease in handling. Appendixes must be lettered alphabetically (A, B, etc.).

Data Item Description
1. Title Software Requirements Specification *2. Identification numbers* ENG-005
3. Description/Purpose The purpose of this document, or set of documentation, is to define the Software Requirements Specification for the R4 HR application.
4. Issue Date As Required Update as required *5. Office of Primary Interest* Technical Authority, copy to Contract Authority *6. GIDEP Applicable*
7. Applicable References & Interrelationship CDRL Item 2005 Annex A, Statement of Work, Section 7.0
8. Originator Contractor *9. Applicable forms*
10. Preparation instructions The Software Requirements Specification must be written in Contractor's format. The Software Requirements Specification must cover, but is not limited to, the following topics: ***1. Scope.*** This section must be divided into the following paragraphs. 1.1 Identification. This paragraph must contain a full identification of the system and the software to which this document applies, including, as applicable, identification number(s), title(s), abbreviation(s), version number(s), and release number(s). 1.2 System overview. This paragraph must briefly state the purpose of the system and the software to which this document applies. It must describe the general nature of the system and software; summarize the history of system development, operation, and maintenance; identify the project sponsor, acquirer, user, developer, and support agencies; identify current and planned operating sites; and list other relevant documents. 1.3 Document overview. This paragraph must summarize the purpose and contents of this document and must describe any security or privacy considerations associated with its use.

2. Referenced documents.

This section must list the number, title, revision, and date of all documents referenced in this specification. This section must also identify the source for all documents not available through normal Government stocking activities.

3. Requirements.

This section must be divided into the following paragraphs to specify the CSCI requirements, that is, those characteristics of the CSCI that are conditions for its acceptance. CSCI requirements are software requirements generated to satisfy the system requirements allocated to this CSCI. Each requirement must be assigned a project-unique identifier to support testing and traceability and must be stated in such a way that an objective test can be defined for it. Each requirement must be annotated with associated qualification method(s) (see section 4) and traceability to system (or subsystem, if applicable) requirements (see section 5.a) if not provided in those sections. The degree of detail to be provided must be guided by the following rule: Include those characteristics of the CSCI that are conditions for CSCI acceptance; defer to design descriptions those characteristics that the acquirer is willing to leave up to the developer. If there are no requirements in a given paragraph, the paragraph must so state. If a given requirement fits into more than one paragraph, it may be stated once and referenced from the other paragraphs.

3.1 Required states and modes.

If the CSCI is required to operate in more than one state or mode having requirements distinct from other states or modes, this paragraph must identify and define each state and mode. Examples of states and modes include: idle, ready, active, post-use analysis, training, degraded, emergency, backup, wartime, peacetime. The distinction between states and modes is arbitrary. A CSCI may be described in terms of states only, modes only, states within modes, modes within states, or any other scheme that is useful. If no states or modes are required, this paragraph must so state, without the need to create artificial distinctions. If states and/or modes are required, each requirement or group of requirements in this specification must be correlated to the states and modes. The correlation may be indicated by a table or other method in this paragraph, in an appendix referenced from this paragraph or by annotation of the requirements in the paragraphs where they appear.

3.2 CSCI capability requirements.

This paragraph must be divided into subparagraphs to itemize the requirements associated with each capability of the CSCI. A "capability" is defined as a group of related requirements. The word "capability" may be replaced with "function," "subject," "object," or other term useful for presenting the requirements.

3.2.x (CSCI capability).

This paragraph must identify a required CSCI capability and must itemize the requirements associated with the capability. If the capability can be more clearly specified by dividing it into constituent capabilities, the constituent capabilities must be specified in subparagraphs. The requirements must specify required behavior of the CSCI and must include applicable parameters, such as response times, throughput times, other timing constraints, sequencing, accuracy, capacities (how much/how many), priorities, continuous operation requirements, and allowable deviations based on operating conditions. The requirements must include, as applicable, required behavior under unexpected, unallowed, or "out of bounds" conditions, requirements for error handling, and any provisions to be incorporated into the CSCI to provide continuity of operations in the event of emergencies. Paragraph 3.3.x of this DID provides a list of topics to be considered when specifying requirements regarding inputs the CSCI must accept and outputs it must produce.

3.3 CSCI external interface requirements.

This paragraph must be divided into subparagraphs to specify the requirements, if any, for the CSCI's external interfaces. This paragraph may reference one or more Interface Requirements Specifications (IRSs) or other documents containing these requirements.

3.3.1 Interface identification and diagrams.

This paragraph must identify the required external interfaces of the CSCI (that is, relationships with other entities that involve sharing, providing or exchanging data). The identification of each interface must include a project-unique identifier and must designate the interfacing entities (systems, configuration items, users, etc.) by name, number, version, and documentation references, as applicable. The identification must state which entities have fixed interface characteristics (and therefore impose interface requirements on interfacing entities) and which are being developed or modified (thus having interface requirements imposed on them). One or more interface diagrams must be provided to depict the interfaces.

3.3.x (Project-unique identifier of interface).

This paragraph (beginning with 3.3.2) must identify a CSCI external interface by project-unique identifier, must briefly identify the interfacing entities, and must be divided into subparagraphs as needed to state the requirements imposed on the CSCI to achieve the interface. Interface characteristics of the other entities involved in the interface must be stated as assumptions or as "When [the entity not covered] does this, the CSCI shall...," not as requirements on the other entities. This paragraph may reference other documents (such as data dictionaries, standards for communication protocols, and standards for user interfaces) in place of stating the information here. The requirements must include the following, as applicable, presented in any order suited to the requirements, and must note any differences in these characteristics from the point of view of the interfacing entities (such as different expectations about the size, frequency, or other characteristics of data elements):
a. Priority that the CSCI must assign the interface
b. Requirements on the type of interface (such as real-time data transfer, storage-and-retrieval of data, etc.) to be implemented
c. Required characteristics of individual data elements that the CSCI must provide, store, send, access, receive, etc., such as:

1) Names/identifiers

a) Project-unique identifier;

b) Non-technical (natural-language) name;

c) DoD standard data element name;

d) Technical name (e.g., variable or field name in code or database); and

e) Abbreviation or synonymous names.

2) Data type (alphanumeric, integer, etc.);

3) Size and format (such as length and punctuation of a character string);

4) Units of measurement (such as meters, dollars, nanoseconds);

5) Range or enumeration of possible values (such as 0-99);

6) Accuracy (how correct) and precision (number of significant digits);

7) Priority, timing, frequency, volume, sequencing, and other constraints, such as whether the data element may be updated and whether business rules apply;

8) Security and privacy constraints; and

9) Sources (setting/sending entities) and recipients (using/receiving entities).

d. Required characteristics of data element assemblies (records, messages, files, arrays, displays, reports, etc.) that the CSCI must provide, store, send, access, receive, etc., such as:

- 1) Names/identifiers

    a) Project-unique identifier;

    b) Non-technical (natural language) name;

    c) Technical name (e.g., record or data structure name in code or database);

    d) Abbreviations or synonymous names.

2) Data elements in the assembly and their structure (number, order, grouping);

3) Medium (such as disk) and structure of data elements/assemblies on the medium;

4) Visual and auditory characteristics of displays and other outputs (such as colors, layouts, fonts, icons and other display elements, beeps, lights);

5) Relationships among assemblies, such as sorting/access characteristics;

6) Priority, timing, frequency, volume, sequencing, and other constraints, such as whether the assembly may be updated and whether business rules apply;

7) Security and privacy constraints; and

8) Sources (setting/sending entities) and recipients (using/receiving entities)

e. Required characteristics of communication methods that the CSCI must use for the interface, such as:

1) Project-unique identifier(s);

2) Communication links/bands/frequencies/media and their characteristics;

3) Message formatting;

4) Flow control (such as sequence numbering and buffer allocation);

5) Data transfer rate, whether periodic/aperiodic, and interval between transfers;

6) Routing, addressing, and naming conventions;

7) Transmission services, including priority and grade; and

8) Safety/security/privacy considerations, such as encryption, user authentication, compartmentalization, and auditing.

f. Required characteristics of protocols the CSCI must use for the interface, such as:

1) Project-unique identifier(s);

2) Priority/layer of the protocol;

3) Packeting, including fragmentation and reassembly, routing, and addressing;

4) Legality checks, error control, and recovery procedures;

5) Synchronization, including connection establishment, maintenance, termination; and

6) Status, identification, and any other reporting features.

g. Other required characteristics, such as physical compatibility of the interfacing entities (dimensions, tolerances, loads, plug compatibility, etc.), voltages, etc.

3.4 CSCI internal interface requirements.

This paragraph must specify the requirements, if any, imposed on interfaces internal to the CSCI. If all internal interfaces are left to the design, this fact must be so stated. If such requirements are to be imposed, paragraph 3.3 of this DID provides a list of topics to be considered.

3.5 CSCI internal data requirements.

This paragraph must specify the requirements, if any, imposed on data internal to the CSCI. Included must be requirements, if any, on databases and data files to be included in the CSCI. If all decisions about internal data are left to the design, this fact must be so stated. If such requirements are to be imposed, paragraphs 3.3.x.c and 3.3.x.d of this DID provide a list of topics to be considered.

3.6 Adaptation requirements.

This paragraph must specify the requirements, if any, concerning installation-dependent data to be provided by the CSCI (such as site-dependent latitude and longitude or site-dependent state tax codes) and operational parameters that the CSCI is required to use that may vary according to operational needs (such as parameters indicating operation-dependent targeting constants or data recording).

3.7 Security and privacy requirements.

This paragraph must specify the CSCI requirements, if any, concerned with maintaining security and privacy. These requirements must include, as applicable, the security/privacy environment in which the CSCI must operate, the type and degree of security or privacy to be provided, the security/privacy risks the CSCI must withstand, required safeguards to reduce those risks, the security/privacy policy that must be met, the security/privacy accountability the CSCI must provide, and the criteria that must be met for security/privacy certification/accreditation.

3.8 CSCI environment requirements.

This paragraph must specify the requirements, if any, regarding the environment in which the CSCI must operate. Examples include the computer hardware and operating system on which the CSCI must run. (Additional requirements concerning computer resources are given in the next paragraph.)

3.9 Computer resource requirements.

This paragraph must be divided into the following subparagraphs.

3.9.1 Computer hardware requirements.

This paragraph must specify the requirements, if any, regarding computer hardware that must be used by the CSCI. The requirements must include, as applicable, number of each type of equipment, type, size, capacity, and other required characteristics of processors, memory, input/output devices, auxiliary storage, communications/network equipment, and other required equipment.

3.9.2 Computer hardware resource utilization requirements.

This paragraph must specify the requirements, if any, on the CSCI's computer hardware resource utilization, such as maximum allowable use of processor capacity, memory capacity, input/output device capacity, auxiliary storage device capacity, and communications/network equipment capacity. The requirements (stated, for example, as percentages of the capacity of each computer hardware resource) must include the conditions, if any, under which the resource utilization is to be measured.

3.9.3 Computer software requirements.

This paragraph must specify the requirements, if any, regarding computer software that must be used by, or incorporated into, the CSCI. Examples include operating systems, database management systems, communications/ network software, utility software, input and equipment simulators, test software, and

manufacturing software. The correct nomenclature, version, and documentation references of each such software item must be provided.

3.9.4 Computer communications requirements.

This paragraph must specify the additional requirements, if any, concerning the computer communications that must be used by the CSCI. Examples include geographic locations to be linked; configuration and network topology; transmission techniques; data transfer rates; gateways; required system use times; type and volume of data to be transmitted/received; time boundaries for transmission/ reception/response; peak volumes of data; and diagnostic features.

3.10 Software quality factors.

This paragraph must specify the CSCI requirements, if any, concerned with software quality factors identified in the contract or derived from a higher level specification. Examples include quantitative requirements regarding CSCI functionality (the ability to perform all required functions), reliability (the ability to perform with correct, consistent results), maintainability (the ability to be easily corrected), availability (the ability to be accessed and operated when needed), flexibility (the ability to be easily adapted to changing requirements), portability (the ability to be easily modified for a new environment), reusability (the ability to be used in multiple applications), testability (the ability to be easily and thoroughly tested), usability (the ability to be easily learned and used), and other attributes.

3.11 Design and implementation constraints.

This paragraph must specify the requirements, if any, that constrain the design and implementation of the CSCI. These requirements may be specified by reference to appropriate commercial or military standards and specifications. Examples include requirements concerning:
a. Use of a particular CSCI architecture or requirements on the architecture, such as required databases or other software units; use of standard, military, or existing components; or use of Government/acquirer-furnished property (equipment, information, or software)
b. Use of particular design or implementation standards; use of particular data standards; use of a particular programming language
c. Flexibility and expandability that must be provided to support anticipated areas of growth or changes in technology, threat, or mission

3.12 Training-related requirements.

This paragraph must specify the CSCI requirements, if any, pertaining to training. Examples include training software to be included in the CSCI.

3.13 Logistics-related requirements.

This paragraph must specify the CSCI requirements, if any, concerned with logistics considerations. These considerations may include: system maintenance, software support, system transportation modes, supply system requirements, impact on existing facilities, and impact on existing equipment.

3.14 Other requirements.

This paragraph must specify additional CSCI requirements, if any, not covered in the previous paragraphs.

3.15 Packaging requirements.

This section must specify the requirements, if any, for packaging, labeling, and handling the CSCI for delivery (for example, delivery on 8 track magnetic tape labelled and packaged in a certain way). Applicable military specifications and standards may be referenced if appropriate.

3.16 Precedence and criticality of requirements.

This paragraph must specify, if applicable, the order of precedence, criticality, or assigned weights indicating the relative importance of the requirements in this specification. Examples include identifying

those requirements deemed critical to safety, to security, or to privacy for purposes of singling them out for special treatment. If all requirements have equal weight, this paragraph must so state.

4. Qualification provisions.

This section must define a set of qualification methods and must specify for each requirement in Section 3 the method(s) to be used to ensure that the requirement has been met. A table may be used to present this information, or each requirement in Section 3 may be annotated with the method(s) to be used. Qualification methods may include:
a. Demonstration: The operation of the CSCI, or a part of the CSCI, that relies on observable functional operation not requiring the use of instrumentation, special test equipment, or subsequent analysis.
b. Test: The operation of the CSCI, or a part of the CSCI, using instrumentation or other special test equipment to collect data for later analysis.
c. Analysis: The processing of accumulated data obtained from other qualification methods. Examples are reduction, interpretation, or extrapolation of test results.
d. Inspection: The visual examination of CSCI code, documentation, etc.
e. Special qualification methods: Any special qualification methods for the CSCI, such as special tools, techniques, procedures, facilities, and acceptance limits.

5. Requirements traceability.

This paragraph must contain:
a. Traceability from each CSCI requirement in this specification to the system (or subsystem, if applicable) requirements it addresses. (Alternatively, this traceability may be provided by annotating each requirement in Section 3.)
Note: Each level of system refinement may result in requirements not directly traceable to higher-level requirements. For example, a system architectural design that creates multiple CSCIs may result in requirements about how the CSCIs will interface, even though these interfaces are not covered in system requirements. Such requirements may be traced to a general requirement such as "system implementation" or to the system design decisions that resulted in their generation.
b. Traceability from each system (or subsystem, if applicable) requirement allocated to this CSCI to the CSCI requirements that address it. All system (subsystem) requirements allocated to this CSCI must be accounted for. Those that trace to CSCI requirements contained in IRSs must reference those IRSs.

6. Notes.

This section must contain any general information that aids in understanding this specification (e.g., background information, glossary, rationale). This section must include an alphabetical listing of all acronyms, abbreviations, and their meanings as used in this document and a list of any terms and definitions needed to understand this document.

A. Appendixes.

Appendixes may be used to provide information published separately for convenience in document maintenance (e.g., charts, classified data). As applicable, each appendix must be referenced in the main body of the document where the data would normally have been provided. Appendixes may be bound as separate documents for ease in handling. Appendixes must be lettered alphabetically (A, B, etc.).

Data Item Description
1. Title Interface Requirements Specification *2. Identification numbers* ENG-006
3. Description/Purpose The purpose of this document, or set of documentation, is to define the Interface Requirements Specification for the R4 HR application.
4. Issue Date As Required Update as required *5. Office of Primary Interest* Technical Authority, copy to Contract Authority *6. GIDEP Applicable*
7. Applicable References & Interrelationship CDRL Item 2006 Annex A, Statement of Work, Section 7.0
8. Originator Contractor *9. Applicable forms*
10. Preparation instructions The Interface Requirements Specification must be written in Contractor's format. The Interface Requirements Specification must cover, but is not limited to, the following topics: ***1. Scope.*** This section must be divided into the following paragraphs. 1.1 Identification. This paragraph must contain a full identification of the systems, the interfacing entities, and the interfaces to which this document applies, including, as applicable, identification number(s), title(s), abbreviation(s), version number(s), and release number(s). 1.2 System overview. This paragraph must briefly state the purpose of the system(s) and software to which this document applies. It must describe the general nature of the system and software; summarize the history of system development, operation, and maintenance; identify the project sponsor, acquirer, user, developer, and support agencies; identify current and planned operating sites; and list other relevant documents. 1.3 Document overview. This paragraph must summarize the purpose and contents of this document and must describe any security or privacy considerations associated with its use.

2. Referenced documents.

This section must list the number, title, revision, and date of all documents referenced in this specification. This section must also identify the source for all documents not available through normal Government stocking activities.

3. Requirements.

This section must be divided into the following paragraphs to specify the requirements imposed on one or more systems, subsystems, configuration items, manual operations, or other system components to achieve one or more interfaces among these entities. Each requirement must be assigned a project-unique identifier to support testing and traceability and must be stated in such a way that an objective test can be defined for it. Each requirement must be annotated with associated qualification method(s) (see section 4) and traceability to system (or subsystem, if applicable) requirements (see section 5.a) if not provided in those sections. The degree of detail to be provided must be guided by the following rule: Include those characteristics of the interfacing entities that are conditions for their acceptance; defer to design descriptions those characteristics that the acquirer is willing to leave up to the developer. If a given requirement fits into more than one paragraph, it may be stated once and referenced from the other paragraphs. If an interfacing entity included in this specification operates in states and/or modes having interface requirements different from other states and modes, each requirement or group of requirements for that entity must be correlated to the states and modes. The correlation may be indicated by a table or other method in this paragraph, in an appendix referenced from this paragraph, or by annotation of the requirements in the paragraphs where they appear.

3.1 Interface identification and diagrams.

For each interface identified in 1.1, this paragraph must include a project-unique identifier and must designate the interfacing entities (systems, configuration items, users, etc.) by name, number, version, and documentation references, as applicable. The identification must state which entities have fixed interface characteristics (and therefore impose interface requirements on interfacing entities) and which are being developed or modified (thus having interface requirements imposed on them). One or more interface diagrams must be provided to depict the interfaces.

3.x (Project-unique identifier of interface).

This paragraph (beginning with 3.2) must identify an interface by project-unique identifier, must briefly identify the interfacing entities, and must be divided into subparagraphs as needed to state the requirements imposed on one or more of the interfacing entities to achieve the interface. If the interface characteristics of an entity are not covered by this IRS but need to be mentioned to specify the requirements for entities that are, those characteristics must be stated as assumptions or as "When [the entity not covered] does this, the [entity being specified] shall...," rather than as requirements on the entities not covered by this IRS. This paragraph may reference other documents (such as data dictionaries, standards for communication protocols, and standards for user interfaces) in place of stating the information here. The requirements must include the following, as applicable, presented in any order suited to the requirements, and must note any differences in these characteristics from the point of view of the interfacing entities (such as different expectations about the size, frequency, or other characteristics of data elements):
a. Priority that the interfacing entity(ies) must assign the interface
b. Requirements on the type of interface (such as real-time data transfer, storage-and-retrieval of data, etc.) to be implemented
c. Required characteristics of individual data elements that the interfacing entity(ies) must provide, store, send, access, receive, etc., such as:

- 1) Names/identifiers

    a) Project-unique identifier;

    b) Non-technical (natural-language) name;

c) DoD standard data element name;

d) Technical name (e.g., variable or field name in code or database);

e) Abbreviation or synonymous names;

2) Data type (alphanumeric, integer, etc.);

3) Size and format (such as length and punctuation of a character string);

4) Units of measurement (such as meters, dollars, nanoseconds);

5) Range or enumeration of possible values (such as 0-99);

6) Accuracy (how correct) and precision (number of significant digits);

7) Priority, timing, frequency, volume, sequencing, and other constraints, such as whether the data element may be updated and whether business rules apply;

8) Security and privacy constraints; and

9) Sources (setting/sending entities) and recipients (using/receiving entities)

d. Required characteristics of data element assemblies (records, messages, files, arrays, displays, reports, etc.) that the interfacing entity(ies) must provide, store, send, access, receive, etc., such as:

1) Names/identifiers

a) Project-unique identifier;

b) Non-technical (natural language) name;

c) Technical name (e.g., record or data structure name in code or database);

d) Abbreviations or synonymous names

2) Data elements in the assembly and their structure (number, order, grouping);

3) Medium (such as disk) and structure of data elements/assemblies on the medium;

4) Visual and auditory characteristics of displays and other outputs (such as colors, layouts, fonts, icons and other display elements, beeps, lights);

5) Relationships among assemblies, such as sorting/access characteristics;

6) Priority, timing, frequency, volume, sequencing, and other constraints, such as whether the assembly may be updated and whether business rules apply;

7) Security and privacy constraints;

8) Sources (setting/sending entities) and recipients (using/receiving entities)

e. Required characteristics of communication methods that the interfacing entity(ies) must use for the interface, such as:

1) Project-unique identifier(s);

2) Communication links/bands/frequencies/media and their characteristics;

3) Message formatting ;

4) Flow control (such as sequence numbering and buffer allocation);

5) Data transfer rate, whether periodic/aperiodic, and interval between transfers;

6) Routing, addressing, and naming conventions;

7) Transmission services, including priority and grade; and

8) Safety/security/privacy considerations, such as encryption, user authentication, compartmentalization, and auditing.

f. Required characteristics of protocols the interfacing entity(ies) must use for the interface, such as:

1) Project-unique identifier(s);

2) Priority/layer of the protocol;

3) Packeting, including fragmentation and reassembly, routing, and addressing;

4) Legality checks, error control, and recovery procedures;

5) Synchronization, including connection establishment, maintenance, termination;

6) Status, identification, and any other reporting features;

g. Other required characteristics, such as physical compatibility of the interfacing entities (dimensions, tolerances, loads, plug compatibility, etc.), voltages, etc.

3.y Precedence and criticality of requirements.

This paragraph must be numbered as the last paragraph in Section 3 and must specify, if applicable, the order of precedence, criticality, or assigned weights indicating the relative importance of the requirements in this specification. Examples include identifying those requirements deemed critical to safety, to security, or to privacy for purposes of singling them out for special treatment. If all requirements have equal weight, this paragraph must so state.

4. Qualification provisions.

This section must define a set of qualification methods and must specify, for each requirement in Section 3, the qualification method(s) to be used to ensure that the requirement has been met. A table may be used to present this information, or each requirement in Section 3 may be annotated with the method(s) to be used. Qualification methods may include:

a. Demonstration: The operation of interfacing entities that relies on observable functional operation not requiring the use of instrumentation, special test equipment, or subsequent analysis.

b. Test: The operation of interfacing entities using instrumentation or special test equipment to collect data for later analysis.

c. Analysis: The processing of accumulated data obtained from other qualification methods. Examples are reduction, interpretation, or extrapolation of test results.

d. Inspection: The visual examination of interfacing entities, documentation, etc.

e. Special qualification methods: Any special qualification methods for the interfacing entities, such as special tools, techniques, procedures, facilities, and acceptance limits.

5. Requirements traceability.

For system-level interfacing entities, this paragraph does not apply. For each subsystem- or lower-level interfacing entity covered by this IRS, this paragraph must contain:

a. Traceability from each requirement imposed on the entity in this specification to the system (or subsystem, if applicable) requirements it addresses. (Alternatively, this traceability may be provided by annotating each requirement in Section 3.)

Note: Each level of system refinement may result in requirements not directly traceable to higher-level requirements. For example, a system architectural design that creates multiple CSCIs may result in

requirements about how the CSCIs will interface, even though these interfaces are not covered in system requirements. Such requirements may be traced to a general requirement such as "system implementation" or to the system design decisions that resulted in their generation.
b. Traceability from each system (or subsystem, if applicable) requirement that has been allocated to the interfacing entity and that affects an interface covered in this specification to the requirements in this specification that address it.

6. Notes.

This section must contain any general information that aids in understanding this document (e.g., background information, glossary, rationale). This section must include an alphabetical listing of all acronyms, abbreviations, and their meanings as used in this document and a list of any terms and definitions needed to understand this document.

A. Appendixes.

Appendixes may be used to provide information published separately for convenience in document maintenance (e.g., charts, classified data). As applicable, each appendix must be referenced in the main body of the document where the data would normally have been provided. Appendixes may be bound as separate documents for ease in handling. Appendixes must be lettered alphabetically (A, B, etc.).

Data Item Description
1. Title Software Design Description *2. Identification numbers* ENG-007
3. Description/Purpose The purpose of this document, or set of documentation, is to define the Software Design Description for the R4 HR application.
4. Issue Date As Required Update as required *5. Office of Primary Interest* Technical Authority, copy to Contract Authority *6. GIDEP Applicable*
7. Applicable References & Interrelationship CDRL Item 2007 Annex A, Statement of Work, Section 7.0
8. Originator Contractor *9. Applicable forms*
10. Preparation instructions The Software Design Description must be written in Contractor's format. The Software Design Description must cover, but is not limited to, the following topics: ***1. Scope.*** This section must be divided into the following paragraphs. 1.1 Identification. This paragraph must contain a full identification of the system and the software to which this document applies, including, as applicable, identification number(s), title(s), abbreviation(s), version number(s), and release number(s). 1.2 System overview. This paragraph must briefly state the purpose of the system and the software to which this document applies. It must describe the general nature of the system and software; summarize the history of system development, operation, and maintenance; identify the project sponsor, acquirer, user, developer, and support agencies; identify current and planned operating sites; and list other relevant documents. 1.3 Document overview. This paragraph must summarize the purpose and contents of this document and must describe any security or privacy considerations associated with its use.

2. Referenced documents.

This section must list the number, title, revision, and date of all documents referenced in this document. This section must also identify the source for all documents not available through normal Government stocking activities.

3. CSCI-wide design decisions.

This section must be divided into paragraphs as needed to present CSCI-wide design decisions, that is, decisions about the CSCI's behavioral design (how it will behave, from a user's point of view, in meeting its requirements, ignoring internal implementation) and other decisions affecting the selection and design of the software units that make up the CSCI. If all such decisions are explicit in the CSCI requirements or are deferred to the design of the CSCI's software units, this section must so state. Design decisions that respond to requirements designated critical, such as those for safety, security, or privacy, must be placed in separate subparagraphs. If a design decision depends upon system states or modes, this dependency must be indicated. Design conventions needed to understand the design must be presented or referenced. Examples of CSCI-wide design decisions are the following:
a. Design decisions regarding inputs the CSCI will accept and outputs it will produce, including interfaces with other systems, HWCIs, CSCIs, and users (4.3.x of this DID identifies topics to be considered in this description). If part or all of this information is given in Interface Design Descriptions (IDDs), they may be referenced.
b. Design decisions on CSCI behavior in response to each input or condition, including actions the CSCI will perform, response times and other performance characteristics, description of physical systems modeled, selected equations/algorithms/rules, and handling of unallowed inputs or conditions.
c. Design decisions on how databases/data files will appear to the user (4.3.x of this DID identifies topics to be considered in this description). If part or all of this information is given in Database Design Descriptions (DBDDs), they may be referenced.
d. Selected approach to meeting safety, security, and privacy requirements.
e. Other CSCI-wide design decisions made in response to requirements, such as selected approach to providing required flexibility, availability, and maintainability.

4. CSCI architectural design.

This section must be divided into the following paragraphs to describe the CSCI architectural design. If part or all of the design depends upon system states or modes, this dependency must be indicated. If design information falls into more than one paragraph, it may be presented once and referenced from the other paragraphs. Design conventions needed to understand the design must be presented or referenced.

4.1 CSCI components.

This paragraph must:
a. Identify the software units that make up the CSCI. Each software unit must be assigned a project-unique identifier.
Note: A software unit is an element in the design of a CSCI; for example, a major subdivision of a CSCI, a component of that subdivision, a class, object, module, function, routine, or database. Software units may occur at different levels of a hierarchy and may consist of other software units. Software units in the design may or may not have a one-to-one relationship with the code and data entities (routines, procedures, databases, data files, etc.) that implement them or with the computer files containing those entities. A database may be treated as a CSCI or as a software unit. The SDD may refer to software units by any name(s) consistent with the design methodology being used.
b. Show the static (such as "consists of") relationship(s) of the software units. Multiple relationships may be presented, depending on the selected software design methodology (for example, in an object-oriented design, this paragraph may present the class and object structures as well as the module and process architectures of the CSCI).
c. State the purpose of each software unit and identify the CSCI requirements and CSCI-wide design decisions allocated to it. (Alternatively, the allocation of requirements may be provided in 6.a.)
d. Identify each software unit's development status/type (such as new development, existing design or software to be reused as is, existing design or software to be reengineered, software to be developed for

reuse, software planned for Build N, etc.) For existing design or software, the description must provide identifying information, such as name, version, documentation references, library, etc.
e. Describe the CSCI's (and as applicable, each software unit's) planned utilization of computer hardware resources (such as processor capacity, memory capacity, input/output device capacity, auxiliary storage capacity, and communications/network equipment capacity). The description must cover all computer hardware resources included in resource utilization requirements for the CSCI, in system-level resource allocations affecting the CSCI, and in resource utilization measurement planning in the Software Development Plan. If all utilization data for a given computer hardware resource are presented in a single location, such as in one SDD, this paragraph may reference that source. Included for each computer hardware resource must be:

1) The CSCI requirements or system-level resource allocations being satisfied;

2) The assumptions and conditions on which the utilization data are based (for example, typical usage, worst-case usage, assumption of certain events);

3) Any special considerations affecting the utilization (such as use of virtual memory, overlays, or multiprocessors or the impacts of operating system overhead, library software, or other implementation overhead);

4) The units of measure used (such as percentage of processor capacity, cycles per second, bytes of memory, kilobytes per second); and

5) The level(s) at which the estimates or measures will be made (such as software unit, CSCI, or executable program) f. Identify the program library in which the software that implements each software unit is to be placed

4.2 Concept of execution.

This paragraph must describe the concept of execution among the software units. It must include diagrams and descriptions showing the dynamic relationship of the software units, that is, how they will interact during CSCI operation, including, as applicable, flow of execution control, data flow, dynamically controlled sequencing, state transition diagrams, timing diagrams, priorities among units, handling of interrupts, timing/sequencing relationships, exception handling, concurrent execution, dynamic allocation/de-allocation, dynamic creation/deletion of objects, processes, tasks, and other aspects of dynamic behavior.

4.3 Interface design.

This paragraph must be divided into the following subparagraphs to describe the interface characteristics of the software units. It must include both interfaces among the software units and their interfaces with external entities such as systems, configuration items, and users. If part or all of this information is contained in Interface Design Descriptions (IDDs), in section 5 of the SDD, or elsewhere, these sources may be referenced.

4.3.1 Interface identification and diagrams.

This paragraph must state the project-unique identifier assigned to each interface and must identify the interfacing entities (software units, systems, configuration items, users, etc.) by name, number, version, and documentation references, as applicable. The identification must state which entities have fixed interface characteristics (and therefore impose interface requirements on interfacing entities) and which are being developed or modified (thus having interface requirements imposed on them). One or more interface diagrams must be provided, as appropriate, to depict the interfaces.

4.3.x (Project-unique identifier of interface).

This paragraph (beginning with 4.3.2) must identify an interface by project-unique identifier, must briefly identify the interfacing entities, and must be divided into subparagraphs as needed to describe the interface characteristics of one or both of the interfacing entities. If a given interfacing entity is not covered by this SDD (for example, an external system) but its interface characteristics need to be mentioned to

describe interfacing entities that are, these characteristics must be stated as assumptions or as "When [the entity not covered] does this, [the entity that is covered] will ...." This paragraph may reference other documents (such as data dictionaries, standards for protocols, and standards for user interfaces) in place of stating the information here. The design description must include the following, as applicable, presented in any order suited to the information to be provided, and must note any differences in these characteristics from the point of view of the interfacing entities (such as different expectations about the size, frequency, or other characteristics of data elements):
a. Priority assigned to the interface by the interfacing entity(ies)
b. Type of interface (such as real-time data transfer, storage-and-retrieval of data, etc.) to be implemented

c. Characteristics of individual data elements that the interfacing entity(ies) will provide, store, send, access, receive, etc., such as:

1) Names/identifiers;

a) Project-unique identifier;

b) Non-technical (natural-language) name ;

c) DoD standard data element name;

d) Technical name (e.g., variable or field name in code or database);

e) Abbreviation or synonymous names;

2) Data type (alphanumeric, integer, etc.);

3) Size and format (such as length and punctuation of a character string);

4) Units of measurement (such as meters, dollars, nanoseconds);

5) Range or enumeration of possible values (such as 0-99);

6) Accuracy (how correct) and precision (number of significant digits);

7) Priority, timing, frequency, volume, sequencing, and other constraints, such as whether the data element may be updated and whether business rules apply;

8) Security and privacy constraints;

9) Sources (setting/sending entities) and recipients (using/receiving entities)

d. Characteristics of data element assemblies (records, messages, files, arrays, displays, reports, etc.) that the interfacing entity(ies) will provide, store, send, access, receive, etc., such as:

1) Names/identifiers

a) Project-unique identifier;

b) Non-technical (natural language) name;

c) Technical name (e.g., record or data structure name in code or database)

d) Abbreviations or synonymous names;

2) Data elements in the assembly and their structure (number, order, grouping);

3) Medium (such as disk) and structure of data elements/assemblies on the medium;

4) Visual and auditory characteristics of displays and other outputs (such as colors, layouts, fonts, icons and other display elements, beeps, lights);

5) Relationships among assemblies, such as sorting/access characteristics;

6) Priority, timing, frequency, volume, sequencing, and other constraints, such as whether the assembly may be updated and whether business rules apply;

7) Security and privacy constraints;

8) Sources (setting/sending entities) and recipients (using/receiving entities);

e. Characteristics of communication methods that the interfacing entity(ies) will use for the interface, such as:

1) Project-unique identifier(s);

2) Communication links/bands/frequencies/media and their characteristics;

3) Message formatting;

4) Flow control (such as sequence numbering and buffer allocation);

5) Data transfer rate, whether periodic/aperiodic, and interval between transfers;

6) Routing, addressing, and naming conventions;

7) Transmission services, including priority and grade; and

8) Safety/security/privacy considerations, such as encryption, user authentication, compartmentalization, and auditing

f. Characteristics of protocols that the interfacing entity(ies) will use for the interface, such as:

1) Project-unique identifier(s);

2) Priority/layer of the protocol;

3) Packeting, including fragmentation and reassembly, routing, and addressing;

4) Legality checks, error control, and recovery procedures;

5) Synchronization, including connection establishment, maintenance, termination; and

6) Status, identification, and any other reporting features;

g. Other characteristics, such as physical compatibility of the interfacing entity(ies) (dimensions, tolerances, loads, voltages, plug compatibility, etc.)

5. CSCI detailed design.

This section must be divided into the following paragraphs to describe each software unit of the CSCI. If part of all of the design depends upon system states or modes, this dependency must be indicated. If design information falls into more than one paragraph, it may be presented once and referenced from the other paragraphs. Design conventions needed to understand the design must be presented or referenced. Interface characteristics of software units may be described here, in Section 4, or in Interface Design Descriptions (IDDs). Software units that are databases, or that are used to access or manipulate databases, may be described here or in Database Design Descriptions (DBDDs).

5.x (Project-unique identifier of a software unit, or designator of a group of software units).

This paragraph must identify a software unit by project-unique identifier and must describe the unit. The description must include the following information, as applicable. Alternatively, this paragraph may designate a group of software units and identify and describe the software units in subparagraphs. Software units that contain other software units may reference the descriptions of those units rather than repeating information.

a. Unit design decisions, if any, such as algorithms to be used, if not previously selected

b. Any constraints, limitations, or unusual features in the design of the software unit

c. The programming language to be used and rationale for its use if other than the specified CSCI language

d. If the software unit consists of or contains procedural commands (such as menu selections in a database management system (DBMS) for defining forms and reports, on-line DBMS queries for database access and manipulation, input to a graphical user interface (GUI) builder for automated code generation, commands to the operating system, or shell scripts), a list of the procedural commands and reference to user manuals or other documents that explain them

e. If the software unit contains, receives, or outputs data, a description of its inputs, outputs, and other data elements and data element assemblies, as applicable. Paragraph 4.3.x of this DID provides a list of topics to be covered, as applicable. Data local to the software unit must be described separately from data input to or output from the software unit. If the software unit is a database, a corresponding Database Design Description (DBDD) must be referenced; interface characteristics may be provided here or by referencing section 4 or the corresponding Interface Design Description(s).

f. If the software unit contains logic, the logic to be used by the software unit, including, as applicable:

1) Conditions in effect within the software unit when its execution is initiated;

2) Conditions under which control is passed to other software units;

3) Response and response time to each input, including data conversion, renaming, and data transfer operations;

4) Sequence of operations and dynamically controlled sequencing during the software unit's operation, including:

a) The method for sequence control;

b) The logic and input conditions of that method, such as timing variations, priority assignments;

c) Data transfer in and out of memory; and

d) The sensing of discrete input signals, and timing relationships between interrupt operations within the software unit.

5) Exception and error handling

6. Requirements traceability.

This section must contain:

a. Traceability from each software unit identified in this SDD to the CSCI requirements allocated to it. (Alternatively, this traceability may be provided in 4.1.)

b. Traceability from each CSCI requirement to the software units to which it is allocated.

7. Notes.

This section must contain any general information that aids in understanding this document (e.g., background information, glossary, rationale). This section must include an alphabetical listing of all

acronyms, abbreviations, and their meanings as used in this document and a list of any terms and definitions needed to understand this document.

A. Appendixes.

Appendixes may be used to provide information published separately for convenience in document maintenance (e.g., charts, classified data). As applicable, each appendix must be referenced in the main body of the document where the data would normally have been provided. Appendixes may be bound as separate documents for ease in handling. Appendixes must be lettered alphabetically (A, B, etc.).

Data Item Description
1. Title Interface Design Description *2. Identification numbers* ENG-008
3. Description/Purpose The purpose of this document, or set of documentation, is to define the Interface Design Description for the R4 HR application.
4. Issue Date As Required Update as required *5. Office of Primary Interest* Technical Authority, copy to Contract Authority *6. GIDEP Applicable*
7. Applicable References & Interrelationship CDRL Item 2008 Annex A, Statement of Work, Section 7.0
8. Originator Contractor *9. Applicable forms*
10. Preparation instructions The Interface Design Description must be written in Contractor's format. The Interface Design Description must cover, but is not limited to, the following topics: ***1. Scope.*** This section must be divided into the following paragraphs. 1.1 Identification. This paragraph must contain a full identification of the system(s), the interfacing entities, and interfaces to which this document applies, including, as applicable, identification number(s), title(s), abbreviation(s), version number(s), and release number(s). 1.2 System overview. This paragraph must briefly state the purpose of the system(s) and software to which this document applies. It must describe the general nature of the system and software; summarize the history of system development, operation, and maintenance; identify the project sponsor, acquirer, user, developer, and support agencies; identify current and planned operating sites; and list other relevant documents. 1.3 Document overview. This paragraph must summarize the purpose and contents of this document and must describe any security or privacy considerations associated with its use.

2. Referenced documents.

This section must list the number, title, revision, and date of all documents referenced in this document. This section must also identify the source for all documents not available through normal Government stocking activities.

3. Interface design.

This section must be divided into the following paragraphs to describe the interface characteristics of one or more systems, subsystems, configuration items, manual operations, or other system components. If part or all of the design depends upon system states or modes, this dependency must be indicated. If design information falls into more than one paragraph, it may be presented once and referenced from the other paragraphs. If part or all of this information is documented elsewhere, it may be referenced. Design conventions needed to understand the design must be presented or referenced.

3.1 Interface identification and diagrams.

For each interface identified in 1.1, this paragraph must state the project-unique identifier assigned to the interface and must identify the interfacing entities (systems, configuration items, users, etc.) by name, number, version, and documentation references, as applicable. The identification must state which entities have fixed interface characteristics (and therefore impose interface requirements on interfacing entities) and which are being developed or modified (thus having interface requirements imposed on them). One or more interface diagrams must be provided, as appropriate, to depict the interfaces.

3.x (Project-unique identifier of interface).

This paragraph (beginning with 3.2) must identify an interface by project-unique identifier, must briefly identify the interfacing entities, and must be divided into subparagraphs as needed to describe the interface characteristics of one or both of the interfacing entities. If a given interfacing entity is not covered by this IDD (for example, an external system) but its interface characteristics need to be mentioned to describe interfacing entities that are, these characteristics must be stated as assumptions or as "When [the entity not covered] does this, [the entity that is covered] will ...." This paragraph may reference other documents (such as data dictionaries, standards for protocols, and standards for user interfaces) in place of stating the information here. The design description must include the following, as applicable, presented in any order suited to the information to be provided, and must note any differences in these characteristics from the point of view of the interfacing entities (such as different expectations about the size, frequency, or other characteristics of data elements):
a. Priority assigned to the interface by the interfacing entity(ies)
b. Type of interface (such as real-time data transfer, storage-and-retrieval of data, etc.) to be implemented

c. Characteristics of individual data elements that the interfacing entity(ies) will provide, store, send, access, receive, etc., such as:

1) Names/identifiers

a) Project-unique identifier

b) Non-technical (natural-language) name

c) DoD standard data element name

d) Technical name (e.g., variable or field name in code or database)

e) Abbreviation or synonymous names

2) Data type (alphanumeric, integer, etc.)

3) Size and format (such as length and punctuation of a character string)

4) Units of measurement (such as meters, dollars, nanoseconds)

5) Range or enumeration of possible values (such as 0-99)

6) Accuracy (how correct) and precision (number of significant digits)

7) Priority, timing, frequency, volume, sequencing, and other constraints, such as whether the data element may be updated and whether business rules apply

8) Security and privacy constraints

9) Sources (setting/sending entities) and recipients (using/receiving entities)

d. Characteristics of data element assemblies (records, messages, files, arrays, displays, reports, etc.) that the interfacing entity(ies) will provide, store, send, access, receive, etc., such as:

1) Names/identifiers

a) Project-unique identifier

b) Non-technical (natural language) name

c) Technical name (e.g., record or data structure name in code or database)

d) Abbreviations or synonymous names

2) Data elements in the assembly and their structure (number, order, grouping)

3) Medium (such as disk) and structure of data elements/assemblies on the medium

4) Visual and auditory characteristics of displays and other outputs (such as colors, layouts, fonts, icons and other display elements, beeps, lights)

5) Relationships among assemblies, such as sorting/access characteristics

6) Priority, timing, frequency, volume, sequencing, and other constraints, such as whether the assembly may be updated and whether business rules apply

7) Security and privacy constraints

8) Sources (setting/sending entities) and recipients (using/receiving entities)

e. Characteristics of communication methods that the interfacing entity(ies) will use for the interface, such as:

1) Project-unique identifier(s)

2) Communication links/bands/frequencies/media and their characteristics

3) Message formatting

4) Flow control (such as sequence numbering and buffer allocation)

5) Data transfer rate, whether periodic/aperiodic, and interval between transfers

6) Routing, addressing, and naming conventions

7) Transmission services, including priority and grade

8) Safety/security/privacy considerations, such as encryption, user authentication, compartmentalization, and auditing

f. Characteristics of protocols the interfacing entity(ies) will use for the interface, such as:

1) Project-unique identifier(s)

2) Priority/layer of the protocol

3) Packeting, including fragmentation and reassembly, routing, and addressing

4) Legality checks, error control, and recovery procedures

5) Synchronization, including connection establishment, maintenance, termination

6) Status, identification, and any other reporting features

g. Other characteristics, such as physical compatibility of the interfacing entity(ies) (dimensions, tolerances, loads, voltages, plug compatibility, etc.)

4. Requirements traceability.

This paragraph must contain: a. Traceability from each interfacing entity covered by this IDD to the system or CSCI requirements addressed by the entity's interface design. b. Traceability from each system or CSCI requirement that affects an interface covered in this IDD to the interfacing entities that address it.

5. Notes.

This section must contain any general information that aids in understanding this document (e.g., background information, glossary, rationale). This section must include an alphabetical listing of all acronyms, abbreviations, and their meanings as used in this document and a list of any terms and definitions needed to understand this document.

A. Appendixes.

Appendixes may be used to provide information published separately for convenience in document maintenance (e.g., charts, classified data). As applicable, each appendix must be referenced in the main body of the document where the data would normally have been provided. Appendixes may be bound as separate documents for ease in handling. Appendixes must be lettered alphabetically (A, B, etc.).

Data Item Description
1. Title Database Design Description *2. Identification numbers* ENG-009
3. Description/Purpose The purpose of this document is, or set of documentation, is to define the Database Design Description for the R4 HR application.
4. Issue Date As Required Update as required *5. Office of Primary Interest* Technical Authority, copy to Contract Authority *6. GIDEP Applicable*
7. Applicable References & Interrelationship CDRL Item 2009 Annex A, Statement of Work, Section 7.0
8. Originator Contractor *9. Applicable forms*
10. Preparation instructions The Database Design Description must be written in Contractor's format. The Database Design Description must cover, but is not limited to, the following topics: ***1. Scope.*** This section must be divided into the following paragraphs. 1.1 Identification. This paragraph must contain a full identification of the database to which this document applies, including, as applicable, identification number(s), title(s), abbreviation(s), version number(s), and release number(s). 1.2 Database overview. This paragraph must briefly state the purpose of the database to which this document applies. It must describe the general nature of the database; summarize the history of its development, use, and maintenance; identify the project sponsor, acquirer, user, developer, and support agencies; identify current and planned operating sites; and list other relevant documents. 1.3 Document overview. This paragraph must summarize the purpose and contents of this document and must describe any security or privacy considerations associated with its use.

2. Referenced documents.

This section must list the number, title, revision, and date of all documents referenced in this manual. This section must also identify the source for all documents not available through normal Government stocking activities.

3. Database-wide design decisions.

This section must be divided into paragraphs as needed to present database-wide design decisions, that is, decisions about the database's behavioral design (how it will behave, from a user's point of view, in meeting its requirements, ignoring internal implementation) and other decisions affecting further design of the database. If all such decisions are explicit in the system or CSCI requirements, this section must so state. Design decisions that respond to requirements designated critical, such as those for safety, security, or privacy, must be placed in separate subparagraphs. If a design decision depends upon system states or modes, this dependency must be indicated. If some or all of the design decisions are described in the documentation of a custom or commercial database management system (DBMS), they may be referenced from this section. Design conventions needed to understand the design must be presented or referenced. Examples of database-wide design decisions are the following:
a. Design decisions regarding queries or other inputs the database will accept and outputs (displays, reports, messages, responses, etc.) it will produce, including interfaces with other systems, HWCIs, CSCIs, and users (5.x.d of this DID identifies topics to be considered in this description). If part or all of this information is given in Interface Design Descriptions (IDDs), they may be referenced.
b. Design decisions on database behavior in response to each input or query, including actions, response times and other performance characteristics, selected equations/algorithms/rules, disposition, and handling of unallowed inputs
c. Design decisions on how databases/data files will appear to the user (4.x of this DID identifies topics to be considered in this description)
d. Design decisions on the database management system to be used (including name, version/release) and the type of flexibility to be built into the database for adapting to changing requirements
e. Design decisions on the levels and types of availability, security, privacy, and continuity of operations to be offered by the database
f. Design decisions on database distribution (such as client/server), master database file updates and maintenance, including maintaining consistency, establishing/ reestablishing and maintaining synchronization, enforcing integrity and business rules
g. Design decisions on backup and restoration including data and process distribution strategies, permissible actions during backup and restoration, and special considerations for new or non-standard technologies such as video and sound
h. Design decisions on repacking, sorting, indexing, synchronization, and consistency including automated disk management and space reclamation considerations, optimizing strategies and considerations, storage and size considerations, and population of the database and capture of legacy data

4. Detailed design of the database.

This section must be divided into paragraphs as needed to describe the detailed design of the database. The number of levels of design and the names of those levels must be based on the design methodology used. Examples of database design levels include conceptual, internal, logical, and physical. If part or all of the design depends upon system states or modes, this dependency must be indicated. Design conventions needed to understand the design must be presented or referenced.
Note: This DID uses the term "data element assembly" to mean any entity, relation, schema, field, table, array, etc., that has structure (number/order/grouping of data elements) at a given design level (e.g., conceptual, internal, logical, physical) and the term "data element" to mean any relation, attribute, field, cell, data element, etc. that does not have structure at that level.

4.x (Name of database design level).

This paragraph must identify a database design level and must describe the data elements and data element assemblies of the database in the terminology of the selected design method. The information must include the following, as applicable, presented in any order suited to the information to be provided:

a. Characteristics of individual data elements in the database design, such as:

1) Names/identifiers

a) Project-unique identifier

b) Non-technical (natural-language) name

c) DoD standard data element name

d) Technical name (e.g., field name in the database)

e) Abbreviation or synonymous names

2) Data type (alphanumeric, integer, etc.)

3) Size and format (such as length and punctuation of a character string)

4) Units of measurement (such as meters, dollars, nanoseconds)

5) Range or enumeration of possible values (such as 0-99)

6) Accuracy (how correct) and precision (number of significant digits)

7) Priority, timing, frequency, volume, sequencing, and other constraints, such as whether the data element may be updated and whether business rules apply

8) Security and privacy constraints

9) Sources (setting/sending entities) and recipients (using/receiving entities)

b. Characteristics of data element assemblies (records, messages, files, arrays, displays, reports, etc.) in the database design, such as:

1) Names/identifiers

a) Project-unique identifier

b) Non-technical (natural language) name

c) Technical name (e.g., record or data structure name in code or database)

d) Abbreviations or synonymous names

2) Data elements in the assembly and their structure (number, order, grouping)

3) Medium (such as disk) and structure of data elements/assemblies on the medium

4) Visual and auditory characteristics of displays and other outputs (such as colors, layouts, fonts, icons and other display elements, beeps, lights)

5) Relationships among assemblies, such as sorting/access characteristics

6) Priority, timing, frequency, volume, sequencing, and other constraints, such as whether the assembly may be updated and whether business rules apply

7) Security and privacy constraints

8) Sources (setting/sending entities) and recipients (using/receiving entities)

5. Detailed design of software units used for database access or manipulation.

This section must be divided into the following paragraphs to describe each software unit used for database access or manipulation. If part or all of this information is provided elsewhere, such as in a Software Design Description (SDD), the SDD for a customized DBMS, or the user manual of a commercial DBMS, that information may be referenced rather than repeated here. If part or all of the design depends upon system states or modes, this dependency must be indicated. If design information falls into more than one paragraph, it may be presented once and referenced from the other paragraphs. Design conventions needed to understand the design must be presented or referenced.

5.x (Project-unique identifier of a software unit, or designator for a group of software units).

This paragraph must identify a software unit by project-unique identifier and must describe the unit. The description must include the following information, as applicable. Alternatively, this paragraph may designate a group of software units and identify and describe the software units in subparagraphs. Software units that contain other software units may reference the descriptions of those units rather than repeating information.

a. Unit design decisions, if any, such as algorithms to be used, if not previously selected

b. Any constraints, limitations, or unusual features in the design of the software unit

c. The programming language to be used and rationale for its use if other than the specified CSCI language

d. If the software unit consists of or contains procedural commands (such as menu selections in a database management system (DBMS) for defining forms and reports, on-line DBMS queries for database access and manipulation, input to a graphical user interface (GUI) builder for automated code generation, commands to the operating system, or shell scripts), a list of the procedural commands and a reference to user manuals or other documents that explain them

e. If the software unit contains, receives, or outputs data, a description of its inputs, outputs, and other data elements and data element assemblies, as applicable. Data local to the software unit must be described separately from data input to or output from the software unit. Interface characteristics may be provided here or by referencing Interface Design Description(s). If a given interfacing entity is not covered by this DBDD (for example, an external system) but its interface characteristics need to be mentioned to describe software units that are, these characteristics must be stated as assumptions or as "When [the entity not covered] does this, [the software unit] will...." This paragraph may reference other documents (such as data dictionaries, standards for protocols, and standards for user interfaces) in place of stating the information here. The design description must include the following, as applicable, presented in any order suited to the information to be provided, and must note any differences in these characteristics from the point of view of the interfacing entities (such as different expectations about the size, frequency, or other characteristics of data elements):

1) Project-unique identifier for the interface

2) Identification of the interfacing entities (software units, configuration items, users, etc.) by name, number, version, and documentation references, as applicable

3) Priority assigned to the interface by the interfacing entity(ies)

4) Type of interface (such as real-time data transfer, storage-and-retrieval of data, etc.) to be implemented

5) Characteristics of individual data elements that the interfacing entity(ies) will provide, store, send, access, receive, etc. Paragraph 4.x.a of this DID identifies topics to be covered in this description.

6) Characteristics of data element assemblies (records, messages, files, arrays, displays, reports, etc.) that the interfacing entity(ies) will provide, store, send, access, receive, etc. Paragraph 4.x.b of this DID identifies topics to be covered in this description.

7) Characteristics of communication methods that the interfacing entity(ies) will use for the interface, such as:

a) Project-unique identifier(s)

b) Communication links/bands/frequencies/media and their characteristics

c) Message formatting

d) Flow control (such as sequence numbering and buffer allocation)

e) Data transfer rate, whether periodic/aperiodic, and interval between transfers

f) Routing, addressing, and naming conventions

g) Transmission services, including priority and grade

h) Safety/security/privacy considerations, such as encryption, user authentication, compartmentalization, and auditing

8) Characteristics of protocols that the interfacing entity(ies) will use for the interface, such as:

a) Project-unique identifier(s)

b) Priority/layer of the protocol

c) Packeting, including fragmentation and reassembly, routing, and addressing

d) Legality checks, error control, and recovery procedures

e) Synchronization, including connection establishment, maintenance, termination

f) Status, identification, and any other reporting features

9) Other characteristics, such as physical compatibility of the interfacing entity(ies) (dimensions, tolerances, loads, voltages, plug compatibility, etc.)

f. If the software unit contains logic, the logic to be used by the software unit, including, as applicable:

1) Conditions in effect within the software unit when its execution is initiated

2) Conditions under which control is passed to other software units

3) Response and response time to each input, including data conversion, renaming, and data transfer operations

4) Sequence of operations and dynamically controlled sequencing during the software unit's operation, including:

a) The method for sequence control

b) The logic and input conditions of that method, such as timing variations, priority assignments

c) Data transfer in and out of memory

d) The sensing of discrete input signals, and timing relationships between interrupt operations within the software unit

5) Exception and error handling

6. Requirements traceability.

This section must contain:
a. Traceability from each database or other software unit covered by this DBDD to the system or CSCI requirements it addresses.
b. Traceability from each system or CSCI requirement that has been allocated to a database or other software unit covered in this DBDD to the database or other software units that address it.

7. Notes.

This section must contain any general information that aids in understanding this document (e.g., background information, glossary, rationale). This section must include an alphabetical listing of all acronyms, abbreviations, and their meanings as used in this document and a list of any terms and definitions needed to understand this document.

A. Appendixes.

Appendixes may be used to provide information published separately for convenience in document maintenance (e.g., charts, classified data). As applicable, each appendix must be referenced in the main body of the document where the data would normally have been provided. Appendixes may be bound as separate documents for ease in handling. Appendixes must be lettered alphabetically (A, B, etc.).

Data Item Description
1. Title Software Test Plan *2. Identification numbers* ENG-010
3. Description/Purpose The purpose of this document, or set of documentation, is to define the Software Test Plan for the R4 HR application.
4. Issue Date As Required Update as required *5. Office of Primary Interest* Technical Authority, copy to Contract Authority *6. GIDEP Applicable*
7. Applicable References & Interrelationship CDRL Item 2010 Annex A, Statement of Work, Section 7.0
8. Originator Contractor *9. Applicable forms*
10. Preparation instructions The Software Test Plan must be written in Contractor's format. The Software Test Plan must cover, but is not limited to, the following topics: ***1. Scope.*** This section must be divided into the following paragraphs. 1.1 Identification. This paragraph must contain a full identification of the system and the software to which this document applies, including, as applicable, identification number(s), title(s), abbreviation(s), version number(s), and release number(s). 1.2 System overview. This paragraph must briefly state the purpose of the system and the software to which this document applies. It must describe the general nature of the system and software; summarize the history of system development, operation, and maintenance; identify the project sponsor, acquirer, user, developer, and support agencies; identify current and planned operating sites; and list other relevant documents. 1.3 Document overview. This paragraph must summarize the purpose and contents of this document and must describe any security or privacy considerations associated with its use.

1.4 Relationship to other plans.

This paragraph must describe the relationship, if any, of the STP to related project plans.

2. Referenced documents.

This section must list the number, title, revision, and date of all documents referenced in this plan. This section must also identify the source for all documents not available through normal Government stocking activities.

3. Software test environment.

This section must be divided into the following paragraphs to describe the software test environment at each intended test site. Reference may be made to the Software Development Plan (SDP) for resources that are described there.

3.x (Name of test site(s)).

This paragraph must identify one or more test sites to be used for the testing, and must be divided into the following subparagraphs to describe the software test environment at the site(s). If all tests are conducted at a single site, this paragraph and its subparagraphs must be presented only once. If multiple test sites use the same or similar software test environments, they may be discussed together. Duplicative information among test site descriptions may be reduced by referencing earlier descriptions.

3.x.1 Software items.

This paragraph must identify by name, number, and version, as applicable, the software items (e.g., operating systems, compilers, communications software, related applications software, databases, input files, code auditors, dynamic path analyzers, test drivers, preprocessors, test data generators, test control software, other special test software, postprocessors) necessary to perform the planned testing activities at the test site(s). This paragraph must describe the purpose of each item, describe its media (tape, disk, etc.), identify those that are expected to be supplied by the site, and identify any classified processing or other security or privacy issues associated with the software items.

3.x.2 Hardware and firmware items.

This paragraph must identify by name, number, and version, as applicable, the computer hardware, interfacing equipment, communications equipment, test data reduction equipment, apparatus such as extra peripherals (tape drives, printers, plotters), test message generators, test timing devices, test event records, etc., and firmware items that will be used in the software test environment at the test site(s). This paragraph must describe the purpose of each item, state the period of usage and the number of each item needed, identify those that are expected to be supplied by the site, and identify any classified processing or other security or privacy issues associated with the items.

3.x.3 Other materials.

This paragraph must identify and describe any other materials needed for the testing at the test site(s). These materials may include manuals, software listings, media containing the software to be tested, media containing data to be used in the tests, sample listings of outputs, and other forms or instructions. This paragraph must identify those items that are to be delivered to the site and those that are expected to be supplied by the site. The description must include the type, layout, and quantity of the materials, as applicable. This paragraph must identify any classified processing or other security or privacy issues associated with the items.

3.x.4 Proprietary nature, acquirer's rights, and licensing.

This paragraph must identify the proprietary nature, acquirer's rights, and licensing issues associated with each element of the software test environment.

3.x.5 Installation, testing, and control.

This paragraph must identify the developer's plans for performing each of the following, possibly in conjunction with personnel at the test site(s):
a. Acquiring or developing each element of the software test environment
b. Installing and testing each item of the software test environment prior to its use
c. Controlling and maintaining each item of the software test environment

3.x.6 Participating organizations.

This paragraph must identify the organizations that will participate in the testing at the test sites(s) and the roles and responsibilities of each.

3.x.7 Personnel.

This paragraph must identify the number, type, and skill level of personnel needed during the test period at the test site(s), the dates and times they will be needed, and any special needs, such as multishift operation and retention of key skills to ensure continuity and consistency in extensive test programs.

3.x.8 Orientation plan.

This paragraph must describe any orientation and training to be given before and during the testing. This information must be related to the personnel needs given in 3.1.7. This training may include user instruction, operator instruction, maintenance and control group instruction, and orientation briefings to staff personnel. If extensive training is anticipated, a separate plan may be developed and referenced here.

3.x.9 Tests to be performed.

This paragraph must identify, by referencing section 4, the tests to be performed at the test site(s).

4. Test identification.

This section must be divided into the following paragraphs to identify and describe each test to which this STP applies.

4.1 General information.

This paragraph must be divided into subparagraphs to present general information applicable to the overall testing to be performed.

4.1.1 Test levels.

This paragraph must describe the levels at which testing will be performed, for example, CSCI level or system level.

4.1.2 Test classes.

This paragraph must describe the types or classes of tests that will be performed (for example, timing tests, erroneous input tests, maximum capacity tests).

4.1.3 General test conditions.

This paragraph must describe conditions that apply to all of the tests or to a group of tests. For example: "Each test shall include nominal, maximum, and minimum values;" "each test of type x shall use live data;" "execution size and time shall be measured for each CSCI." Included must be a statement of the extent of testing to be performed and rationale for the extent selected. The extent of testing must be expressed as a percentage of some well defined total quantity, such as the number of samples of discrete operating conditions or values, or other sampling approach. Also included must be the approach to be followed for retesting/regression testing.

4.1.4 Test progression.

In cases of progressive or cumulative tests, this paragraph must explain the planned sequence or progression of tests.

4.1.5 Data recording, reduction, and analysis.

This paragraph must identify and describe the data recording, reduction, and analysis procedures to be used during and after the tests identified in this STP. These procedures must include, as applicable, manual, automatic, and semi-automatic techniques for recording test results, manipulating the raw results into a form suitable for evaluation, and retaining the results of data reduction and analysis.

4.2 Planned tests.

This paragraph must be divided into the following subparagraphs to describe the total scope of the planned testing.

4.2.x (Item(s) to be tested).

This paragraph must identify a CSCI, subsystem, system, or other entity by name and project-unique identifier, and must be divided into the following subparagraphs to describe the testing planned for the item(s). (Note: the "tests" in this plan are collections of test cases. There is no intent to describe each test case in this document.)

4.2.x.y (Project-unique identifier of a test).

This paragraph must identify a test by project-unique identifier and must provide the information specified below for the test. Reference may be made as needed to the general information in 4.1.
a. Test objective
b. Test level
c. Test type or class
d. Qualification method(s) as specified in the requirements specification
e. Identifier of the CSCI requirements and, if applicable, software system requirements addressed by this test. (Alternatively, this information may be provided in Section 6.)
f. Special requirements (for example, 48 hours of continuous facility time, weapon simulation, extent of test, use of a special input or database)
g. Type of data to be recorded
h. Type of data recording/reduction/analysis to be employed
i. Assumptions and constraints, such as anticipated limitations on the test due to system or test conditions--timing, interfaces, equipment, personnel, database, etc.
j. Safety, security, and privacy considerations associated with the test

5. Test schedules.

This section must contain or reference the schedules for conducting the tests identified in this plan. It must include:
a. A listing or chart depicting the sites at which the testing will be scheduled and the time frames during which the testing will be conducted
b. A schedule for each test site depicting the activities and events listed below, as applicable, in chronological order with supporting narrative as necessary:

1) Onsite test period and periods assigned to major portions of the testing

2) Pretest onsite period needed for setting up the software test environment and other equipment, system debugging, orientation, and familiarization

3) Collection of database/data file values, input values, and other operational data needed for the testing

4) Conducting the tests, including planned retesting

5) Preparation, review, and approval of the Software Test Report (STR)

6. Requirements traceability.

This paragraph must contain:
a. Traceability from each test identified in this plan to the CSCI requirements and, if applicable, software system requirements it addresses. (Alternatively, this traceability may be provided in 4.2.x.y and referenced from this paragraph.)
b. Traceability from each CSCI requirement and, if applicable, each software system requirement covered by this test plan to the test(s) that address it. The traceability must cover the CSCI requirements in all applicable Software Requirements Specifications (SRSs) and associated Interface Requirements Specifications (IRSs), and, for software systems, the system requirements in all applicable System/ Subsystem Specifications (SSSs) and associated system-level IRSs.

7. Notes.

This section must contain any general information that aids in understanding this document (e.g., background information, glossary, rationale). This section must include an alphabetical listing of all acronyms, abbreviations, and their meanings as used in this document and a list of any terms and definitions needed to understand this document.

A. Appendixes.

Appendixes may be used to provide information published separately for convenience in document maintenance (e.g., charts, classified data). As applicable, each appendix must be referenced in the main body of the document where the data would normally have been provided. Appendixes may be bound as separate documents for ease in handling. Appendixes must be lettered alphabetically (A, B, etc.).

Data Item Description
1. Title Software Test Description *2. Identification numbers* ENG-011
3. Description/Purpose The purpose of this document, or set of documentation, is to define the Software Test Description for the R4 HR application.
4. Issue Date As Required Update as required *5. Office of Primary Interest* Technical Authority, copy to Contract Authority *6. GIDEP Applicable*
7. Applicable References & Interrelationship CDRL Item 2011 Annex A, Statement of Work, Section 7.0
8. Originator Contractor *9. Applicable forms*
10. Preparation instructions The Software Test Description must be written in Contractor's format. The Software Test Description must cover, but is not limited to, the following topics: ***1. Scope.*** This section must be divided into the following paragraphs. 1.1 Identification. This paragraph must contain a full identification of the system and the software to which this document applies, including, as applicable, identification number(s), title(s), abbreviation(s), version number(s), and release number(s). 1.2 System overview. This paragraph must briefly state the purpose of the system and the software to which this document applies. It must describe the general nature of the system and software; summarize the history of system development, operation, and maintenance; identify the project sponsor, acquirer, user, developer, and support agencies; identify current and planned operating sites; and list other relevant documents. 1.3 Document overview. This paragraph must summarize the purpose and contents of this document and must describe any security or privacy considerations associated with its use.

2. Referenced documents.

This section must list the number, title, revision, and date of all documents referenced in this document. This section must also identify the source for all documents not available through normal Government stocking activities.

3. Test preparations.

This section must be divided into the following paragraphs. Safety precautions, marked by WARNING or CAUTION, and security and privacy considerations must be included as applicable.

3.x (Project-unique identifier of a test).

This paragraph must identify a test by project-unique identifier, must provide a brief description, and must be divided into the following subparagraphs. When the information required duplicates information previously specified for another test, that information may be referenced rather than repeated.

3.x.1 Hardware preparation.

This paragraph must describe the procedures necessary to prepare the hardware for the test. Reference may be made to published operating manuals for these procedures. The following must be provided, as applicable:
a. The specific hardware to be used, identified by name and, if applicable, number;
b. Any switch settings and cabling necessary to connect the hardware;
c. One or more diagrams to show hardware, interconnecting control, and data paths; and
d. Step-by-step instructions for placing the hardware in a state of readiness.

3.x.2 Software preparation.

This paragraph must describe the procedures necessary to prepare the item(s) under test and any related software, including data, for the test. Reference may be made to published software manuals for these procedures. The following information must be provided, as applicable:
a. The specific software to be used in the test;
b. The storage medium of the item(s) under test (e.g., magnetic tape, diskette);
c. The storage medium of any related software (e.g., simulators, test drivers, databases);
d. Instructions for loading the software, including required sequence;
e. Instructions for software initialization common to more than one test case.

3.x.3 Other pretest preparations.

This paragraph must describe any other pre-test personnel actions, preparations, or procedures necessary to perform the test.

4. Test descriptions.

This section must be divided into the following paragraphs. Safety precautions, marked by WARNING or CAUTION, and security and privacy considerations must be included as applicable.

4.x (Project unique identifier of a test).

This paragraph must identify a test by project unique identifier and must be divided into the following subparagraphs. When the required information duplicates information previously provided, that information may be referenced rather than repeated.

4.x.y (Project-unique identifier of a test case).

This paragraph must identify a test case by project-unique identifier, state its purpose, and provide a brief description. The following subparagraphs must provide a detailed description of the test case.

4.x.y.1 Requirements addressed.

This paragraph must identify the CSCI or system requirements addressed by the test case. (Alternatively, this information may be provided in 5.a.)

4.x.y.2 Prerequisite conditions.

This paragraph must identify any prerequisite conditions that must be established prior to performing the test case. The following considerations must be discussed, as applicable:
a. Hardware and software configuration;
b. Flags, initial breakpoints, pointers, control parameters, or initial data to be set/reset prior to test commencement;
c. Preset hardware conditions or electrical states necessary to run the test case;
d. Initial conditions to be used in making timing measurements;
e. Conditioning of the simulated environment; and
f. Other special conditions peculiar to the test case;

4.x.y.3 Test inputs.

This paragraph must describe the test inputs necessary for the test case. The following must be provided, as applicable:
a. Name, purpose, and description (e.g., range of values, accuracy) of each test input;
b. Source of the test input and the method to be used for selecting the test input;
c. Whether the test input is real or simulated;
d. Time or event sequence of test input;
e. The manner in which the input data will be controlled to:

1) Test the item(s) with a minimum/reasonable number of data types and values

2) Exercise the item(s) with a range of valid data types and values that test for overload, saturation, and other "worst case" effects

3) Exercise the item(s) with invalid data types and values to test for appropriate handling of irregular inputs

4) Permit retesting, if necessary

4.x.y.4 Expected test results.

This paragraph must identify all expected test results for the test case. Both intermediate and final test results must be provided, as applicable.

4.x.y.5 Criteria for evaluating results.

This paragraph must identify the criteria to be used for evaluating the intermediate and final results of the test case. For each test result, the following information must be provided, as applicable:
a. The range or accuracy over which an output can vary and still be acceptable
b. Minimum number of combinations or alternatives of input and output conditions that constitute an acceptable test result
c. Maximum/minimum allowable test duration, in terms of time or number of events
d. Maximum number of interrupts, halts, or other system breaks that may occur
e. Allowable severity of processing errors
f. Conditions under which the result is inconclusive and retesting is to be performed
g. Conditions under which the outputs are to be interpreted as indicating irregularities in input test data, in the test database/data files, or in test procedures
h. Allowable indications of the control, status, and results of the test and the readiness for the next test case (may be output of auxiliary test software)
i. Additional criteria not mentioned above.

4.x.y.6 Test procedure.

This paragraph must define the test procedure for the test case. The test procedure must be defined as a series of individually numbered steps listed sequentially in the order in which the steps are to be performed. For convenience in document maintenance, the test procedures may be included as an appendix and referenced in this paragraph. The appropriate level of detail in each test procedure depends on the type of software being tested. For some software, each keystroke may be a separate test procedure step; for most software, each step may include a logically related series of keystrokes or other actions. The appropriate level of detail is the level at which it is useful to specify expected results and compare them to actual results. The following must be provided for each test procedure, as applicable:
a. Test operator actions and equipment operation required for each step, including commands, as applicable, to:

1) Initiate the test case and apply test inputs

2) Inspect test conditions

3) Perform interim evaluations of test results

4) Record data

5) Halt or interrupt the test case

6) Request data dumps or other aids, if needed

7) Modify the database/data files

8) Repeat the test case if unsuccessful

9) Apply alternate modes as required by the test case

10) Terminate the test case

b. Expected result and evaluation criteria for each step
c. If the test case addresses multiple requirements, identification of which test procedure step(s) address which requirements. (Alternatively, this information may be provided in 5.)
d. Actions to follow in the event of a program stop or indicated error, such as:

1) Recording of critical data from indicators for reference purposes

2) Halting or pausing time-sensitive test-support software and test apparatus

3) Collection of system and operator records of test results

e. Procedures to be used to reduce and analyze test results to accomplish the following, as applicable:

1) Detect whether an output has been produced

2) Identify media and location of data produced by the test case

3) Evaluate output as a basis for continuation of test sequence

4) Evaluate test output against required output

4.x.y.7 Assumptions and constraints.

This paragraph must identify any assumptions made and constraints or limitations imposed in the description of the test case due to system or test conditions, such as limitations on timing, interfaces, equipment, personnel, and database/data files. If waivers or exceptions to specified limits and parameters are approved, they must be identified and this paragraph must address their effects and impacts upon the test case.

5. Requirements traceability.

This paragraph must contain:
a. Traceability from each test case in this STD to the system or CSCI requirements it addresses. If a test case addresses multiple requirements, traceability from each set of test procedure steps to the requirement(s) addressed. (Alternatively, this traceability may be provided in 4.x.y.1.)
b. Traceability from each system or CSCI requirement covered by this STD to the test case(s) that address it. For CSCI testing, traceability from each CSCI requirement in the CSCI's Software Requirements Specification (SRS) and associated Interface Requirements Specifications (IRSs). For system testing, traceability from each system requirement in the system's System/Subsystem Specification (SSS) and associated IRSs. If a test case addresses multiple requirements, the traceability must indicate the particular test procedure steps that address each requirement.

6. Notes.

This section must contain any general information that aids in understanding this document (e.g., background information, glossary, rationale). This section must include an alphabetical listing of all acronyms, abbreviations, and their meanings as used in this document and a list of any terms and definitions needed to understand this document.

A. Appendixes.

Appendixes may be used to provide information published separately for convenience in document maintenance (e.g., charts, classified data). As applicable, each appendix must be referenced in the main body of the document where the data would normally have been provided. Appendixes may be bound as separate documents for ease in handling. Appendixes must be lettered alphabetically (A, B, etc.).

Data Item Description
1. Title Software Test Report *2. Identification numbers* ENG-012
3. Description/Purpose The purpose of this document, or set of documentation, is to define the Software Test Report for the R4 HR application.
4. Issue Date As Required Update as required *5. Office of Primary Interest* Technical Authority, copy to Contract Authority *6. GIDEP Applicable*
7. Applicable References & Interrelationship CDRL Item 2012 Annex A, Statement of Work, Section 7.0
8. Originator Contractor *9. Applicable forms*
10. Preparation instructions The Software Test Report must be written in Contractor's format. The Software Test Report must cover, but is not limited to, the following topics: ***1. Scope.*** This section must be divided into the following paragraphs. 1.1 Identification. This paragraph must contain a full identification of the system and the software to which this document applies, including, as applicable, identification number(s), title(s), abbreviation(s), version number(s), and release number(s). 1.2 System overview. This paragraph must briefly state the purpose of the system and the software to which this document applies. It must describe the general nature of the system and software; summarize the history of system development, operation, and maintenance; identify the project sponsor, acquirer, user, developer, and support agencies; identify current and planned operating sites; and list other relevant documents. 1.3 Document overview. This paragraph must summarize the purpose and contents of this document and must describe any security or privacy considerations associated with its use.

2. Referenced documents.

This section must list the number, title, revision, and date of all documents referenced in this report. This section must also identify the source for all documents not available through normal Government stocking activities.

3. Overview of test results.

This section must be divided into the following paragraphs to provide an overview of test results.

3.1 Overall assessment of the software tested.

This paragraph must:
a. Provide an overall assessment of the software as demonstrated by the test results in this report
b. Identify any remaining deficiencies, limitations, or constraints that were detected by the testing performed. Problem/change reports may be used to provide deficiency information.
c. For each remaining deficiency, limitation, or constraint, describe:

1) Its impact on software and system performance, including identification of requirements not met

2) The impact on software and system design to correct it

3) A recommended solution/approach for correcting it

3.2 Impact of test environment.

This paragraph must provide an assessment of the manner in which the test environment may be different from the operational environment and the effect of this difference on the test results.

3.3 Recommended improvements.

This paragraph must provide any recommended improvements in the design, operation, or testing of the software tested. A discussion of each recommendation and its impact on the software may be provided. If no recommended improvements are provided, this paragraph must state "None."

4. Detailed test results.

This section must be divided into the following paragraphs to describe the detailed results for each test. Note: The word "test" means a related collection of test cases.

4.x (Project-unique identifier of a test).

This paragraph must identify a test by project-unique identifier and must be divided into the following subparagraphs to describe the test results.

4.x.1 Summary of test results.

This paragraph must summarize the results of the test. The summary must include, possibly in a table, the completion status of each test case associated with the test (for example, "all results as expected," "problems encountered," "deviations required"). When the completion status is not "as expected," this paragraph must reference the following paragraphs for details.

4.x.2 Problems encountered.

This paragraph must be divided into subparagraphs that identify each test case in which one or more problems occurred.

4.x.2.y (Project-unique identifier of a test case).

This paragraph must identify by project-unique identifier a test case in which one or more problems occurred, and must provide:
a. A brief description of the problem(s) that occurred

b. Identification of the test procedure step(s) in which they occurred
c. Reference(s) to the associated problem/change report(s) and backup data, as applicable
d. The number of times the procedure or step was repeated in attempting to correct the problem(s) and the outcome of each attempt
e. Backup points or test steps where tests were resumed for retesting

4.x.3 Deviations from test cases/procedures.

This paragraph must be divided into subparagraphs that identify each test case in which deviations from test case/test procedures occurred.

4.x.3.y (Project-unique identifier of a test case).

This paragraph must identify by project-unique identifier a test case in which one or more deviations occurred, and must provide:
a. A description of the deviation(s) (for example, test case run in which the deviation occurred and nature of the deviation, such as substitution of required equipment, procedural steps not followed, schedule deviations). (Red-lined test procedures may be used to show the deviations)
b. The rationale for the deviation(s)
c. An assessment of the deviations' impact on the validity of the test case

5. Test log.

This section must present, possibly in a figure or appendix, a chronological record of the test events covered by this report. This test log must include:
a. The date(s), time(s), and location(s) of the tests performed
b. The hardware and software configurations used for each test including, as applicable, part/model/serial number, manufacturer, revision level, and calibration date of all hardware, and version number and name for the software components used
c. The date and time of each test-related activity, the identity of the individual(s) who performed the activity, and the identities of witnesses, as applicable

6. Notes.

This section must contain any general information that aids in understanding this document (e.g., background information, glossary, rationale). This section must include an alphabetical listing of all acronyms, abbreviations, and their meanings as used in this document and a list of any terms and definitions needed to understand this document.

A. Appendixes.

Appendixes may be used to provide information published separately for convenience in document maintenance (e.g., charts, classified data). As applicable, each appendix must be referenced in the main body of the document where the data would normally have been provided. Appendixes may be bound as separate documents for ease in handling. Appendixes must be lettered alphabetically (A, B, etc.).

Data Item Description
1. Title Software User Manual *2. Identification numbers* ENG-013
3. Description/Purpose The purpose of this document, or set of documentation, is to define the Software User Manual for the R4 HR application.
4. Issue Date As Required Update as required *5. Office of Primary Interest* Technical Authority, copy to Contract Authority *6. GIDEP Applicable*
7. Applicable References & Interrelationship CDRL Item 2013 Annex A, Statement of Work, Section 7.0
8. Originator Contractor *9. Applicable forms*
10. Preparation instructions The Software User Manual must be written in Contractor's format. The Software User Manual must cover, but is not limited to, the following topics: ***1. Scope.*** This section must be divided into the following paragraphs. 1.1 Identification. This paragraph must contain a full identification of the system and the software to which this document applies, including, as applicable, identification number(s), title(s), abbreviation(s), version number(s), and release number(s). 1.2 System overview. This paragraph must briefly state the purpose of the system and the software to which this document applies. It must describe the general nature of the system and software; summarize the history of system development, operation, and maintenance; identify the project sponsor, acquirer, user, developer, and support agencies; identify current and planned operating sites; and list other relevant documents. 1.3 Document overview. This paragraph must summarize the purpose and contents of this manual and must describe any security or privacy considerations associated with its use.

2. Referenced documents.

This section must list the number, title, revision, and date of all documents referenced in this manual. This section must also identify the source for all documents not available through normal Government stocking activities.

3. Software summary.

This section must be divided into the following paragraphs.

3.1 Software application.

This paragraph must provide a brief description of the intended uses of the software. Capabilities, operating improvements, and benefits expected from its use must be described.

3.2 Software inventory.

This paragraph must identify all software files, including databases and data files that must be installed for the software to operate. The identification must include security and privacy considerations for each file and identification of the software necessary to continue or resume operation in case of an emergency.

3.3 Software environment.

This paragraph must identify the hardware, software, manual operations, and other resources needed for a user to install and run the software. Included, as applicable, must be identification of:
a. Computer equipment that must be present, including amount of memory needed, amount of auxiliary storage needed, and peripheral equipment such as printers and other input/output devices
b. Communications equipment that must be present
c. Other software that must be present, such as operating systems, databases, data files, utilities, and other supporting systems
d. Forms, procedures, or other manual operations that must be present
e. Other facilities, equipment, or resources that must be present

3.4 Software organization and overview of operation.

This paragraph must provide a brief description of the organization and operation of the software from the user's point of view. The description must include, as applicable:
a. Logical components of the software, from the user's point of view, and an overview of the purpose/operation of each component
b. Performance characteristics that can be expected by the user, such as:

1) Types, volumes, rate of inputs accepted

2) Types, volume, accuracy, rate of outputs that the software can produce

3) Typical response time and factors that affect it

4) Typical processing time and factors that affect it

5) Limitations, such as number of events that can be tracked

6) Error rate that can be expected

7) Reliability that can be expected

c. Relationship of the functions performed by the software with interfacing systems, organizations, or positions
d. Supervisory controls that can be implemented (such as passwords) to manage the software

3.5 Contingencies and alternate states and modes of operation.

This paragraph must explain differences in what the user will be able to do with the software at times of emergency and in various states and modes of operation, if applicable.

3.6 Security and privacy.

This paragraph must contain an overview of the security and privacy considerations associated with the software. A warning must be included regarding making unauthorized copies of software or documents, if applicable.

3.7 Assistance and problem reporting.

This paragraph must identify points of contact and procedures to be followed to obtain assistance and report problems encountered in using the software.

4. Access to the software.

This section must contain step-by-step procedures oriented to the first time/occasional user. Enough detail must be presented so that the user can reliably access the software before learning the details of its functional capabilities. Safety precautions, marked by WARNING or CAUTION, must be included where applicable.

4.1 First-time user of the software.

This paragraph must be divided into the following subparagraphs.

4.1.1 Equipment familiarization.

This paragraph must describe the following as appropriate:
a. Procedures for turning on power and making adjustments
b. Dimensions and capabilities of the visual display screen
c. Appearance of the cursor, how to identify an active cursor if more than one cursor can appear, how to position a cursor, and how to use a cursor
d. Keyboard layout and role of different types of keys and pointing devices
e. Procedures for turning power off if special sequencing of operations is needed

4.1.2 Access control.

This paragraph must present an overview of the access and security features of the software that are visible to the user. The following items must be included, as applicable:
a. How and from whom to obtain a password
b. How to add, delete, or change passwords under user control
c. Security and privacy considerations pertaining to the storage and marking of output reports and other media that the user will generate

4.1.3 Installation and setup.

This paragraph must describe any procedures that the user must perform to be identified or authorized to access or install software on the equipment, to perform the installation, to configure the software, to delete or overwrite former files or data, and to enter parameters for software operation.

4.2 Initiating a session.

This paragraph must provide step-by-step procedures for beginning work, including any options available. A checklist for problem determination must be included in case difficulties are encountered.

4.3 Stopping and suspending work.

This paragraph must describe how the user can cease or interrupt use of the software and how to determine whether normal termination or cessation has occurred.

5. Processing reference guide.

This section must provide the user with procedures for using the software. If procedures are complicated or extensive, additional Sections 6, 7, ... may be added in the same paragraph structure as this section and with titles meaningful to the sections selected. The organization of the document will depend on the characteristics of the software being documented. For example, one approach is to base the sections on the organizations in which users work, their assigned positions, their work sites, or the tasks they must perform. For other software, it may be more appropriate to have Section 5 be a guide to menus, Section 6 be a guide to the command language used, and Section 7 be a guide to functions. Detailed procedures are intended to be presented in subparagraphs of paragraph 5.3. Depending on the design of the software, the subparagraphs might be organized on a function-by-function, menu-by-menu, transaction-by-transaction, or other basis. Safety precautions, marked by WARNING or CAUTION, must be included where applicable.

5.1 Capabilities.

This paragraph must briefly describe the interrelationships of the transactions, menus, functions, or other processes in order to provide an overview of the use of the software.

5.2 Conventions.

This paragraph must describe any conventions used by the software, such as the use of colors in displays, the use of audible alarms, the use of abbreviated vocabulary, and the use of rules for assigning names or codes.

5.3 Processing procedures.

This paragraph must explain the organization of subsequent paragraphs, e.g., by function, by menu, by screen. Any necessary order in which procedures must be accomplished must be described.

5.3.x (Aspect of software use).

The title of this paragraph must identify the function, menu, transaction, or other process being described. This paragraph must describe and give options and examples, as applicable, of menus, graphical icons, data entry forms, user inputs, inputs from other software or hardware that may affect the software's interface with the user, outputs, diagnostic or error messages or alarms, and help facilities that can provide on-line descriptive or tutorial information. The format for presenting this information can be adapted to the particular characteristics of the software, but a consistent style of presentation must be used, i.e., the descriptions of menus must be consistent, the descriptions of transactions must be consistent among themselves.

5.4 Related processing.

This paragraph must identify and describe any related batch, offline, or background processing performed by the software that is not invoked directly by the user and is not described in paragraph 5.3. Any user responsibilities to support this processing must be specified.

5.5 Data backup.

This paragraph must describe procedures for creating and retaining backup data that can be used to replace primary copies of data in event of errors, defects, malfunctions, or accidents.

5.6 Recovery from errors, malfunctions, and emergencies.

This paragraph must present detailed procedures for restart or recovery from errors or malfunctions occurring during processing and for ensuring continuity of operations in the event of emergencies.

5.7 Messages.

This paragraph must list, or refer to an appendix that lists, all error messages, diagnostic messages, and information messages that can occur while accomplishing any of the user's functions. The meaning of

each message and the action that should be taken after each such message must be identified and described.

5.8 Quick-reference guide.

If appropriate to the software, this paragraph must provide or reference a quick-reference card or page for using the software. This quick-reference guide must summarize, as applicable, frequently-used function keys, control sequences, formats, commands, or other aspects of software use.

6. Notes.

This section must contain any general information that aids in understanding this document (e.g., background information, glossary, rationale). This section must include an alphabetical listing of all acronyms, abbreviations, and their meanings as used in this document and a list of terms and definitions needed to understand this document. If section 5 has been expanded into section(s) 6, ..., this section must be numbered as the next section following section n.

A. Appendixes.

Appendixes may be used to provide information published separately for convenience in document maintenance (e.g., charts, classified data). As applicable, each appendix must be referenced in the main body of the document where the data would normally have been provided. Appendixes may be bound as separate documents for ease in handling. Appendixes must be lettered alphabetically (A, B, etc.).

Data Item Description
1. Title Computer Programming Manual *2. Identification numbers* ENG-014
3. Description/Purpose The purpose of this document, or set of documentation, is to define the Computer Programming Manual for the R4 HR application.
4. Issue Date As Required Update as required *5. Office of Primary Interest* Technical Authority, copy to Contract Authority *6. GIDEP Applicable*
7. Applicable References & Interrelationship CDRL Item 2014 Annex A, Statement of Work, Section 7.0
8. Originator Contractor *9. Applicable forms*
10. Preparation instructions The Computer Programming Manual must be written in Contractor's format. The Computer Programming Manual must cover, but is not limited to, the following topics: ***1. Scope.*** This section must be divided into the following paragraphs. 1.1 Identification. This paragraph must contain the manufacturer's name, model number, and any other identifying information for the computer system to which this document applies. 1.2 Computer system overview. This paragraph must briefly state the purpose of the computer system to which this document applies. 1.3 Document overview. This paragraph must summarize the purpose and contents of this manual and must describe any security or privacy considerations associated with its use. ***2. Referenced documents.*** This section must list the number, title, revision, and date of all documents referenced in this manual. This section must also identify the source for all documents not available through normal Government stocking activities.

3. Programming environment.

This section must be divided into paragraphs as appropriate to provide the following information.
a. The components and configuration of the computer system
b. Operating characteristics, capabilities, and limitations, including, as applicable:

1) Machine cycle time

2) Word length

3) Memory capacity and characteristics

4) Instruction set characteristics

5) Interrupt capabilities

6) Modes of operation (e.g., batch, interactive, privileged, non-privileged)

7) Operational registers

8) Error indicators

9) Input/output characteristics

10) Special features

c. Description of the equipment (e.g., tapes, disks, other peripheral equipment) necessary to perform compilations and assemblies on the computer system. Identify (as applicable) by name and version number the editor, linker, link-editor, compiler, assembler, cross-compilers, cross-assemblers, and other utilities used, and reference appropriate manuals describing their use. Highlight any special flags or instructions necessary for loading, executing, or recording the results.

4. Programming information.

This section must be divided into paragraphs as appropriate to provide the following information.
a. Description of the programming features of the computer's instruction set architecture, including, as applicable:

1) Data representation (e.g., byte, word, integer, floating-point, double precision)

2) Instruction formats and addressing modes

3) Special registers and words (e.g., stack pointer, program counter)

4) Control instructions (e.g., branch, jump, subroutine and procedure call instructions, privileged instructions, and the modes they operate in)

5) Subroutines and procedures (e.g., nonreentrant, reentrant, macrocode routines, argument lists, parameter passing conventions)

6) Interrupt processing

7) Timers and clocks

8) Memory protection features (e.g., read only memory)

9) Additional features, such as instruction or data cache architecture

b. Description of each instruction, including, as applicable:

1) Use

2) Syntax

3) Condition codes set

4) Execution time

5) Machine-code format

6) Mnemonic conventions

7) Other characteristics

c. Description of input and output control programming, including, as applicable:

1) Initial loading and verification of computer memory

2) Serial and parallel data channels

3) Discrete inputs and outputs

4) Interface components

5) Device numbers, operational codes, and memory locations for peripheral equipment

d. Additional, restricted, or special programming techniques associated with the computer system (e.g., a concise description of the micro-program control section)
e. Examples that demonstrate the programming features described above, including examples of the proper use of all categories of instructions on the computer system
f. Error detection and diagnostic features associated with the computer system, including condition codes, overflow and addressing exception interrupts, and input and output error status indicators

5. Notes.

This section must contain any general information that aids in understanding this document (e.g., background information, glossary, rationale). This section must include an alphabetical listing of all acronyms, abbreviations, and their meanings as used in this document and a list of terms and definitions needed to understand this document.

A. Appendixes.

Appendixes may be used to provide information published separately for convenience in document maintenance (e.g., charts, classified data). As applicable, each appendix must be referenced in the main body of the document where the data would normally have been provided. Appendixes may be bound as separate documents for ease in handling. Appendixes must be lettered alphabetically (A, B, etc.).

ANNEX B
Basis of Payment

In the following, Part I refers to the part of work described in Phases 1, 2 and 3; Part II refers to all the remaining work, excluding Phases 1, 2 and 3. The total price will be the sum of the price for Part I and the estimated price for Part II.

The work for Part I must be completed in the first year after contract award, and the work for Part II must be completed in the second and third years after contract award as per the schedule in Section 7.0 of Annex A. If required, work for a one-year option period will follow Part II.

The Bidder must provide, in Part I, Table 1, a firm price for each of the lines (1,2, 3,4) identified in Part I for the first year of the contract.

The firm price must include a price breakdown for Lines 1, 2 and 3 identified in Part I, including the level of effort (number of days), the resource category and the firm all-inclusive per diem rate for the resource category.

Other expenditures include all charges, expenses to acquire equipment or software, or any other fees to support the execution of Part I. The list of other expenditures must be included in the Financial Bid.

The Bidder must provide in Part II, Table 1 a firm per diem rate for the resource categories identified in Part II for years 2 and 3 of the contract.

The Bidder must provide in Option Period, Table 1 a firm per diem rate for the resource categories identified in the Option Period.

The Bidder must provide the total price (sum of the price for Part I and the estimated price for Part II).

Part I (year 1) **Table 1 – Firm Price for the Software Solution**

Line	Description	Total Firm Price	Resource Category	Level of Effort (Number of Days)	All-inclusive Per Diem
1	Phase 1				
	Project/Configuration Management Services				
2	Phase 2				
	Development of System Architecture				
3	Phase 3 (Build 1)				
	Development of Build 1 Version of R4 HR Application				
4	Other expenditures				
	Total (Lines 1-4)				

Part II **Table 1 – Software Refinements Pricing Table**

As and when requested, commencing approximately one year after contract award

Part II (year 2)

Category of Personnel	**Estimated Number of Days**	**Level of Expertise**	**Proposed Firm Per Diem**
Project Manager	50	Level 3	
Software Engineer	100	Level 3	
Simulation engine developer	240	Level 2	
3D GUI programmer	240	Level 2	

/developer			
2D GUI programmer /developer	240	Level 2	
Tester	240	Level 2	
Technical Writer	50	Level 2	

Part II (year 3)

Category of Personnel	Estimated Number of Days	Level of Expertise	Proposed Firm Per Diem
Project Manager	25	Level 3	
Software Engineer	50	Level 3	
Simulation engine developer	120	Level 2	
3D GUI programmer /developer	120	Level 2	
2D GUI programmer /developer	120	Level 2	
Tester	1202	Level 2	
Technical Writer	25	Level 2	

Option Period **Table 1 – Optional Software Refinements Pricing Table**

If required, commencing after completion of Part II.

Category of Personnel	Estimated Number of Days	Level of Expertise	Proposed Firm Per Diem
Project Manager	25	Level 3	
Software Engineer	50	Level 3	
Simulation engine developer	120	Level 2	
3D GUI programmer /developer	120	Level 2	
2D GUI programmer /developer	120	Level 2	
Tester	120	Level 2	
Technical Writer	25	Level 2	

Annex D
Evaluation criteria
Development of R4 HR Application

The bids must contain all relevant information/data for ranking, and information must be clearly located within the bids. The evaluation will be based on five aspects:

- Corporate Experience (20%),
- Project Team Qualifications (20%),
- Technical Approach (15%),
- Project Management (10%), and
- Financial Proposal (35%)

Bid Evaluation Method

The bid evaluation method will be based on a combination of mandatory and point rated criteria.

To be considered, the bidders must a) meet all of the mandatory requirements, and b) meet the minimum requirements under the rated criteria. These criteria are described in the table below. Bids not meeting a) or b) above will be given no further consideration.

Mandatory Criteria

1	**CORPORATE EXPERIENCE** For the purposes of this criterion, corporate experience refers to experience in completing projects similar in scope and size to this requirement. The experience is calculated as follows: completing 1 full- time project of 24 months or 2 full-time projects of 12 months represents two years of experience. The proposal must demonstrate the previous corporate experience of the Bidder in implementing projects in each of the areas specified. Projects under which the experience was obtained are to be identified in relation with the criteria listed below. The Bidder's proposal must detail the role of it in performing the identified project(s) such as whether it had overall or partial responsibility for the project's performance.	Yes/No
1.1	The Bidder must demonstrate that it has a minimum of 36 months in total of recent (within the last 5 years) corporate experience in the design, development and implementation of discrete event simulations.	
1.2	The Bidder must demonstrate that it has a minimum of 36 months in total of recent (within the last 5 years from the date of bid closing) corporate experience in the design, development and implementation of 3D graphical interfaces.	
2	**PROJECT TEAM QUALIFICATIONS**	
2.1	The Bidder must provide a professional résumé for each of the proposed resources. Résumés must include the proposed resource's role and the project name, name of client organization, project description, scope, budget and duration of all related projects.	
2.2	The Bidder must demonstrate the ability to provide fully qualified replacement resources for all proposed primary resources.	
2.3	The programming resources for the development of the simulation engine must have a minimum of 24 months in total of recent (within the last 5 years from the date of bid closing) experience in programming in Python.	

Rated Criteria

The following table describes the rated criteria.

1	**CORPORATE EXPERIENCE (maximum: 20 points – minimum: 12 points)** The information requested in the proposal for these criteria is as described in the Mandatory Criteria above. The Bidder will identify only the projects that pertain to the areas listed at 1.1 and 1.2. Where a single project covers more than one area, it can be identified for evaluation under more than one area. For the purposes of this criterion, corporate experience refers to recent (within the last 5 years from the date of bid closing) experience in completing projects similar in scope and size to this requirement.	Max 20	Min 12
1.1	Corporate experience in the domain of discrete event simulations: – More than 24 months of corporate experience in the creation of the underlying software and not simply the use of discrete event simulation tools (12 points). – More than 12 months of corporate experience in the creation of the underlying software and not simply the use of discrete event simulation tools (10 points). Or, – More than 48 months of corporate experience in the design, development and implementation of discrete event simulations (6 points). – More than 36 months of corporate experience in the design, development and implementation of discrete event simulations (3 points).		
1.2	Corporate experience in the domain of 3D graphical interfaces: – More than 24 months of corporate experience in the design, development and implementation of 3D graphical interfaces supporting near-real-time applications* (8 points). – More than 12 months of corporate experience in the design, development and implementation of 3D graphical interfaces supporting near-real-time applications* (7 points). Or, – More than 48 months of corporate experience in the design, development and implementation of 3D graphical interfaces (5 points). – More than 36 months of corporate experience in the design, development and implementation of 3D graphical interfaces (3 points). ** Near-real-time applications are defined as applications where: a) software events can be scheduled in real time units and they execute approximately at that real time; b) the screen video output is constantly and rapidly updated such that the motion of on-screen objects appears fluid; and c) the software appears to respond immediately to user input (e.g. video games).*		
2	**PROJECT TEAM QUALIFICATIONS (maximum: 20 points – minimum: 12 points)** The information requested in the proposal for these criteria is as described in the Mandatory Criteria above. For the purposes of this criterion, experience refers to the combined recent (within the last 5 years from the date of bid closing) experience of the project team members in completing projects similar in scope and size to this requirement.	20	12
2.1	Project team experience in the design, development and implementation of discrete event simulations: – More than 24 months of experience in the creation of the underlying software and not simply the use of discrete event simulation tools (6 points). – More than 12 months of experience in the creation of the underlying software and not simply the use of discrete event simulation tools (5 points). Or, – More than 24 months of experience in the design, development and implementation of discrete event simulations (3 points).		

	– More than 12 months of experience in the design, development and implementation of discrete event simulations (2 points).		
2.2	Project team experience in using the python programming language: – More than 24 months of experience in the design, development, implementation, and optimization of a python application (6 points). – More than 12 months of experience in the design, development, implementation, and optimization of a python application (5 points). Or, – More than 24 months of experience in the design, development and implementation of a python application (3 points). – More than 12 months of experience in the design, development and implementation of a python application (2 points).		
2.3	Project team experience in the design, development and implementation of 3D graphical interfaces: – More than 24 months of experience in the design, development and implementation of 3D graphical interfaces supporting near-real-time applications* (4 points). – More than 12 months of experience in the design, development and implementation of 3D graphical interfaces supporting near-real-time applications (3 points). Or, – More than 24 months of experience in the design, development and implementation of 3D graphical interfaces (2 points). – More than 12 months of experience in the design, development and implementation of 3D graphical interfaces (1 point). ** Near-real-time applications are defined as applications where: a) software events can be scheduled in real time units and they execute approximately at that real time; b) the screen video output is constantly and rapidly updated such that the motion of on-screen objects appears fluid; and c) the software appears to respond immediately to user input (e.g. video games).*		
2.4	Project team experience in the design, development and implementation of applications making extensive use of a database where the database is integral to the functioning of the application, and in which efficient database usage is critical to the speed of the resulting application: – More than 24 months of experience in the design, development and implementation of applications making extensive use of a database (2 points). – More than 12 months of experience in the design, development and implementation of applications making extensive use of a database (1 point).		
2.5	Project team experience in programming multi-core systems that explicitly make use of parallel processing: – More than 24 months of experience in programming multi-core systems that explicitly make use of parallel processing (2 points). – More than 12 months of experience in programming multi-core systems that explicitly make use of parallel processing (1 point).		
3	**TECHNICAL APPROACH (maximum : 15 points – minimum : 8 points)** The Bidder must demonstrate that it understands the requirement and that it recognizes and can solve the problem areas that could affect the functionality and performance of the R4 HR application. The Bidder must describe the methodology that it will use to develop the R4 HR application.	15	8
3.1	Understanding of requirement: – Complete understanding of the requirement (3 points) – Near complete understanding of the requirement (2 points) – Moderate understanding of the requirement (1 point)		
3.2	Recognition of problem areas: – Comprehensive recognition of problem areas (3 points) – Significant recognition of problem areas (2 points) – Some significant recognition of problem areas (1 point)		

3.3	Solving problem areas: – Provides excellent solutions (6 points) – Provides good solutions (4 points) – Provides acceptable solutions (2 points)		
3.4	Adequacy of software development methodology: – Methodology follows best practices (3 points) – Methodology is acceptable (2 points) – Methodology is lacking in some areas (1 point)		
4	**PROJECT MANAGEMENT (maximum : 10 points – minimum : 5 points)**	10	5
4.1	Experience of Project Manager: – More than five (5) years experience in the last ten (10) years managing software development projects (2 points) – More than two (2) years experience in the last eight (8) years managing software development projects (1 point)		
4.2	Team Organization and Management Approach: The Bidder should describe how it proposes to control the management of the project, including subcontracts. The proposal should clearly demonstrate the project team's organization and reporting structure. This should clearly show the channels of communication between the Technical Authority and the Contractor's project management and team organization. Lines of responsibility within the project team must be identified using a responsibility assignment matrix. Where subcontractors are proposed as part of the project team, the Bidder should provide a list of the individual subcontracts proposed, describe the work to be performed by each one and include/explain the proposed basis of selection for each one. – A complete Responsibility Assignment Matrix (RAM) has been provided which defines and delineates areas of responsibility among corporate members and subcontractors of the Bidder's team. The Project Management approach exceeds requirements for effective performance and administration of the Work. The approach presents little or no potential for disruption in schedule or cost. (2 points) – A Responsibility Assignment Matrix (RAM) has been provided which describes areas of responsibility among corporate members and subcontractors of the Bidder's team, but some areas of responsibility and communication are not fully defined or delineated. The Project Management approach meets requirements for effective performance and administration of the Work. The approach presents little or no potential for disruption in schedule or cost. (1 point)		
4.3	Work Breakdown Structure: The proposal should provide a breakdown of the engineering tasks that will be undertaken for the completion of each of the project milestones as well as tasks related to management activities such as management reporting, oversight, subcontract management, and client liaison that are proposed for completion of each of the milestones. – The Work Breakdown Structure is thorough and complete. It very clearly describes all necessary management, administrative, and engineering activities that are necessary for the successful completion of the work. (2 points) – The Work Breakdown Structure is acceptable but some elements are not fully defined. It clearly describes all necessary management, administrative, and engineering activities that are necessary for the successful completion of the work. (1 point)		
4.4	Project Schedule: The project schedule should include at least planned start, duration, effort and expected dates for each activity.		

	– The project schedule is thorough and complete. It provides time estimates that are reasonable and is organized in a way that maximizes likelihood of delivering on time. (2 points) – The project schedule is acceptable but lacks some detail and/or relevant items. It provides time estimates that are reasonable in most cases and is organized in a way that is likely to deliver on time. (1 point)		
4.5	Risk Identification and Mitigation: The Bidder's proposal should demonstrate a good understanding of all aspects of the work and identify all specific risks which the Bidder anticipates that it will be required to manage. This should address the anticipated technical risks, their management, and how risk and its management may evolve throughout the period of the contract. With the identification of project specific risks, the proposal should also identify what risk-reduction measures will be employed by the Bidder in order to effectively manage the risks. The Bidder should address how it will address the evolution of risk and its management, throughout the Period of the Contract. – The Bidder has identified all of the potential technical risks that may arise in the performance of the work that can be reasonably anticipated at the time of the bidding process. The bidder has thoroughly assessed the evolution of the risks throughout the Contract Period, and has proposed effective risk mitigation strategies in all cases. (2 points) – The Bidder has identified most of the potential technical risks that may arise in the performance of the work that can be reasonably anticipated at the time of the bidding process. The bidder has assessed the evolution of the risks throughout the Contract Period, and has proposed effective risk mitigation strategies in most cases. (1 point)		

Public Works and Government Services Canada

Travaux publics et Services gouvernementaux Canada

RETURN BIDS TO:
RETOURNER LES SOUMISSIONS À:

Bid Receiving - PWGSC / Réception des soumissions - TPSGC
11 Laurier St. / 11, rue Laurier
Place du Portage, Phase III
Core 0A1 / Noyau 0A1
Gatineau
Québec
K1A 0S5
Bid Fax: (819) 997-9776

REQUEST FOR PROPOSAL
DEMANDE DE PROPOSITION

Proposal To: Public Works and Government Services Canada

We hereby offer to sell to Her Majesty the Queen in right of Canada, in accordance with the terms and conditions set out herein, referred to herein or attached hereto, the goods, services, and construction listed herein and on any attached sheets at the price(s) set out therefor.

Proposition aux: Travaux Publics et Services Gouvernementaux Canada

Nous offrons par la présente de vendre à Sa Majesté la Reine du chef du Canada, aux conditions énoncées ou incluses par référence dans la présente et aux annexes ci-jointes, les biens, services et construction énumérés ici sur toute feuille ci-annexée, au(x) prix indiqué(s).

Comments - Commentaires

Title - Sujet SIMULATION ENGINE/GUI DEVELOPMENT	
Solicitation No. - N° de l'invitation W7714-125376/A	**Date** 2013-05-07
Client Reference No. - N° de référence du client W7714-125376	
GETS Reference No. - N° de référence de SEAG	
File No. - N° de dossier 613zm.W7714-125376	**CCC No./N° CCC - FMS No./N° VME**
Solicitation Closes - L'invitation prend fin **at - à 02:00 PM** **on - le 2013-06-14**	**Time Zone** **Fuseau horaire** Eastern Daylight Saving Time EDT
F.O.B. - F.A.B. **Plant-Usine:** ☐ **Destination:** ☐ **Other-Autre:** ☐	
Address Enquiries to: - Adresser toutes questions à: Pierre, Anoule	**Buyer Id - Id de l'acheteur** 613zm
Telephone No. - N° de téléphone (819) 956-2137 ( )	**FAX No. - N° de FAX** (819) 956-1207
Destination - of Goods, Services, and Construction: **Destination - des biens, services et construction:** DEPARTMENT OF NATIONAL DEFENCE NDHQ - DRDC - CORA MAJOR GENERAL GEORGE R. PEARKES BLDG 101 COLONEL BY DRIVE, 8 FLR CBC OTTAWA, ON K1A0K2	

Instructions: See Herein

Instructions: Voir aux présentes

Vendor/Firm Name and Address
Raison sociale et adresse du fournisseur/de l'entrepreneur

Delivery Required - Livraison exigée See Herein	**Delivery Offered - Livraison proposée**
Vendor/Firm Name and Address **Raison sociale et adresse du fournisseur/de l'entrepreneur** **Telephone No. - N° de téléphone** **Facsimile No. - N° de télécopieur**	
Name and title of person authorized to sign on behalf of Vendor/Firm (type or print) **Nom et titre de la personne autorisée à signer au nom du fournisseur/ de l'entrepreneur (taper ou écrire en caractères d'imprimerie)** **Signature** **Date**	

Issuing Office - Bureau de distribution

Informatics Professional Services Division / Division des services professionnels en informatique
11 Laurier St., / 11, rue Laurier
3C2, Place du Portage
Gatineau
Québec
K1A 0S5

Canada

BID SOLICITATION

FOR CONTRACTS AGAINST A SUPPLY ARRANGEMENT FOR

SOLUTION-BASED INFORMATICS PROFESSIONAL SERVICES (SBIPS)

FOR
DEPARTMENT OF NATIONAL DEFENCE

TABLE OF CONTENTS

List of Annexes to the Resulting Contract:

Annex A	Statement of Work
Annex B	Basis of Payment
Annex C	Security Requirement Check List (SRCL)
Annex D	Evaluation Criteria

Forms:

- Form 1 - Bid Submission Form

Solutions-Based Informatics and Professional Services (SBIPS) for the Department of National Defence

PART 1 - GENERAL INFORMATION

1.1 Introduction

This document states terms and conditions that apply to bid solicitation # W7714-125376/A. It is divided into seven parts plus annexes and, if applicable, attachments as follows:

Part 1 General Information: provides a general description of the requirement;

Part 2 Bidder Instructions: provides the instructions, clauses and conditions applicable to the bid solicitation and states that the Bidder agrees to be bound by the clauses and conditions contained in all parts of the bid solicitation;

Part 3 Bid Preparation Instructions: provides bidders with instructions on how to prepare their bid;

Part 4 Evaluation Procedures and Basis of Selection: indicates how the evaluation will be conducted, the evaluation criteria that must be addressed in the bid, if applicable, and the basis of selection;

Part 5 Certifications: includes the certifications to be provided;

Part 6 Security, Financial and Other Requirements: includes specific requirements that must be addressed by bidders; and

Part 7 Resulting Contract Clauses: includes the clauses and conditions that will apply to any resulting contract.

The annexes include the Statement of Work, the Basis of Payment and any other annexes.

1.2 Summary

(a) This bid solicitation is being issued to satisfy the requirement of the Department of National Defence (the "**Client**") for Solutions-based Informatics Professional Services (SBIPS) under the SBIPS Supply Arrangement (SA) method of supply.

(b) It is intended to result in the award of one contract for two and half years, plus one irrevocable one-year option allowing Canada to extend the term of the contract.

(c) There is a security requirement associated with this requirement. For additional information, see Part 6 - Security, Financial and Other Requirements, and Part 7 - Resulting Contract Clauses. Bidders should consult the "Security Requirements on PWGSC Bid Solicitations - Instructions for Bidders" document on the Departmental Standard Procurement Documents (http://www.pwgsc.gc.ca/acquisitions/text/plain/plain-e.html#top) Website.

(d) The requirement is subject to the provisions of the World Trade Organization *Agreement on Government Procurement* (WTO-AGP), the *North American Free Trade Agreement* (NAFTA), the *Agreement on Internal Trade* (AIT), the Canadian-Chile Free Trade Agreement (CCFTA), the Canada-Peru Free Trade Agreement

(CPFTA), the Canada-Columbia Free Trade Agreement (CCo/FTA) and the Canada – Free Trade Agreement Panama (CPan FTA) if it is applicable.

1.3 Description of Requirement

The Department of National Defence (DND) has a requirement for Solution Based informatics Professional Services, for the provision of simulation software (R4 HR Application) that allows analysts to build complex process models, conduct simulations consisting of multiple integrated models, and permit dynamic exploration of the results. The exploitation of this technology will enable the Department of National Defence/Canadian Forces (DND/CF) to respond to more challenging and pressing HR issues in addition to providing the (DND/CF) decision makers with advice based on more comprehensive HR-related analyses.

Only qualified SBIPS SA Holders currently holding a SBIPS SA for Tier 2 in the NCR Region under the EN578-05IT01/G series of SAs are eligible to compete. The SBIPS SA EN578-05IT01/G is incorporated by reference and forms part of this bid solicitation. This SBIPS SA RFP is a requirement involving the following SBIPS Domain of Expertise:

- 11. Systems Integration

PART 2 - BIDDER INSTRUCTIONS

2.1 Standard Instructions, Clauses and Conditions

(a) All instructions, clauses and conditions identified in the bid solicitation by number, date and title are set out in the Standard Acquisition Clauses and Conditions Manual issued by Public Works and Government Services Canada (PWGSC).

(b) Bidders who submit a bid agree to be bound by the instructions, clauses and conditions of the bid solicitation and accept the clauses and conditions of the resulting contract.

(c) The 2003 (2012-11-19) Standard Instructions - Goods or Services - Competitive Requirements are incorporated by reference into and form part of the bid solicitation. If there is a conflict between the provisions of 2003 and this document, this document prevails.

(d) Subsection 5.4 of Standard Instructions - Goods or Services - Competitive Requirements 2003 is amended as follows:

(i) Delete: sixty (60) days

(ii) Insert: 120 days

2.2 Submission of Bids

(a) Bids must be submitted only to Public Works and Government Services Canada Bid Receiving Unit by the date, time and place indicated on page 1 of the bid solicitation.

(b) Due to the nature of the bid solicitation, bids transmitted by facsimile or electronic mail will not be accepted.

(c)Enquiries - Bid Solicitation

(i) All enquiries must be submitted in writing to the Contracting Authority no later than ten calendar days before the bid closing date, unless otherwise specified in each RFP. Enquiries received after that time may not be answered.

(ii) Bidders should reference as accurately as possible the numbered item of the bid solicitation to which the enquiry relates. Care should be taken by bidders to explain each question in sufficient detail in order to enable Canada to provide an accurate answer. Technical enquiries that are of a "proprietary" nature must be clearly marked "proprietary" at each relevant item. Items identified as proprietary will be treated as such except where Canada determines that the enquiry is not of a proprietary nature. Canada may edit the questions or may request that the Bidder do so, so that the proprietary nature of the question is eliminated, and the enquiry can be answered with copies to all bidders. Enquiries not submitted in a form that can be distributed to all bidders may not be answered by Canada.

2.3 Applicable Laws

(a) Any resulting contract must be interpreted and governed, and the relations between the parties determined, by the laws in force in Ontario.

(b) A bidder may, at its discretion, substitute the applicable laws of a Canadian province or territory of their choice without affecting the validity of its bid, by deleting the name of the Canadian province or territory specified and inserting the name of the Canadian province or territory of its choice. If no change is made, it acknowledges that the applicable laws specified are acceptable to the Bidder.

Note to Bidders: Bidders are requested to indicate the Canadian province or territory they wish to apply to any resulting contract in their Bid Submission Form.

2.4 Improvement of Requirement During Solicitation Period

Should bidders consider that the specifications or Statement of Work contained in the bid solicitation could be improved technically or technologically, bidders are invited to make suggestions, in writing, to the Contracting Authority named in the bid solicitation. Bidders must clearly outline the suggested improvement as well as the reasons for the suggestion. Suggestions that do not restrict the level of competition nor favour a particular bidder will be given consideration provided they are submitted to the Contracting Authority in accordance with the article entitled "Enquiries - Bid Solicitation". Canada will have the right to accept or reject any or all suggestions.

2.5 Basis for Canada's Ownership of Intellectual Property

Canada has determined that any intellectual property arising from the performance of the Work under the Contract will belong to Canada, on the following ground:

(a) to augment an existing body of Crown Background as a prerequisite to the transfer of the augmented Background to the private sector, through licensing or assignment of ownership (not necessarily to the original Contractor), for the purposes of Commercial Exploitation; (Article 6.4.2 of the Policy on Title to Intellectual Property Arising Under Crown Procurement Contracts)

(b) The crown will retain the copyright and the right to translation for all source code, software and documentation delivered by the contractor.

2.6 Notes Regarding Background Information

Any pre-existing software that the bidder proposes to incorporate into the R4 HR application must be approved by Canada. Canada will only consider approving software that will not restrict Canada's intention to license the R4 HR technology as outlined below.

Following completion of this contract, Canada intends to license the R4 HR application under two licensing agreements: a research license and a commercial license.

• The research license will be free and will permit the licensee to use and modify the software.

• The commercial license will have a royalty-based fee structure and will grant the licensee the right to use, modify and sell the software. A commercial license may be granted by Canada to a company for the purpose of commercializing and further developing the software and providing services associated with the software including, but not limited to, marketing, distribution, maintenance, upgrades, technical support, training and consulting.

2.7 Volumetric Data

If provided, the data described in this Request for Proposal has been provided to Bidders to assist them in preparing their bids. The inclusion of this data in this bid solicitation does not represent a commitment by Canada that Canada's future usage of the services identified in this request for proposal will be consistent with this data. It is provided purely for information purposes.

PART 3 - BID PREPARATION INSTRUCTIONS

3.1 Bid Preparation Instructions

(a) Unless the RFP specifies otherwise, Canada requests that bidders provide their bid in separately bound sections as follows:

(i) Section I: Technical Bid (4 hard copies and 1 soft copy)

(ii) Section II: Financial Bid (1 hard copy and 1 soft copy)

(iii) Section III: Certifications (2 hard copies)

Where a soft copy is required, if there is a discrepancy between the wording of the soft copy and the hard copy, the wording of the hard copy will have priority over the wording of the soft copy. Prices must appear in the financial bid only. Prices must not be indicated in any other section of the bid.

(b) Canada requests that bidders follow the format instructions described below in the preparation of their bid:

(i) use 8.5 x 11 inch (216 mm x 279 mm) paper;

(ii) use a numbering system that corresponds to the bid solicitation;

(iii) include a title page at the front of each volume of the bid that includes the title, date, bid solicitation number, bidder's name and address and contact information of its representatives; and

(iv) include a table of contents.

(c) **Green Procurement:** In April 2006, Canada issued a policy directing federal departments and agencies to take the necessary steps to incorporate environmental considerations into the procurement process Policy on Green Procurement which can be found at:http://www.tpsgc-pwgsc.gc.ca/ecologisation-greening/achats-procurement/politique-policy-eng.html
To assist Canada in reaching its objectives, bidders are encouraged to:

(i) use paper containing fiber certified as originating from a sustainably- managed forest and/or containing minimum 30% recycled content; and

(ii) use an environmentally-preferable format including black and white printing instead of colour printing, printing double sided/duplex, using staples or clips instead of cerlox, duotangs or binders.

(d) Multiple bids from the same bidder are not permitted in response to this bid solicitation. Each bidder must submit only a single bid. If any bidder submits more than one bid, Canada will choose in its discretion which bid to consider.

3.2A - Section I: Technical Bid

(a) In their technical bid, bidders will demonstrate their understanding of the requirements contained in the bid solicitation and explain how they will meet these requirements. Bidders will demonstrate their capability in a thorough, concise and clear manner for carrying out the work. The technical bid must address clearly and in sufficient depth the points that are subject to the evaluation criteria against which the bid will be evaluated. Simply repeating the statement contained in the bid solicitation is not sufficient. In order to facilitate the evaluation of the bid, Canada requests that bidders address and present topics in the order of the evaluation criteria under the same headings. To avoid duplication, bidders may refer to different sections of their bids by identifying the

specific paragraph and page number where the subject topic has already been addressed. The technical bid consists of the following:

(i) **Bid Submission Form**: Unless specified otherwise in the RFP, Bidders are requested to include the completed Bid Submission Form with their bids. It provides a common form in which bidders can provide information required for evaluation and contract award, such as a contact name, the Bidder's Procurement Business Number, the Bidder's status under the Federal Contractors Program for Employment Equity, etc. Using the form to provide this information is not mandatory, but it is recommended. If Canada determines that the information required by the Bid Submission Form is incomplete or requires correction, Canada will provide the Bidder with an opportunity to do so.

(ii) **Security, Financial & Other Requirements**: As required by Part 6 of the bid solicitation.

(iii) **Résumés for Proposed Resources**: Unless specified otherwise in the RFP, the technical bid must include résumés for the resources identified in the bid solicitation that demonstrate that each proposed individual meets the qualification requirements described (including any educational requirements, work experience requirements, and professional designation or membership requirements). With respect to résumés and resources:

(A) Proposed resources may be employees of the Bidder or employees of a subcontractor, or these individuals may be independent contractors to whom the Bidder would subcontract a portion of the Work. (refer to Part 5, Certifications). For educational requirements for a particular degree, designation or certificate, the Contracting Authority will only consider educational programmes that were successfully completed by the resource by the time of bid closing.

(B) For requirements relating to professional designation or membership, the resource must have the required designation or membership by the time of bid closing and must continue, where applicable, to be a member in good standing of the profession's governing body throughout the evaluation and Contract Period.

(C) For work experience, the Contracting Authority will not consider experience gained as part of an educational programme, except for experience gained through a formal co-operative programme at a post-secondary institution.

(D) For any requirements that specify a particular time period (e.g., 2 years) of work experience, the Contract Authority will disregard any information about experience if the individual's résumé does not include the relevant dates for the experience claimed (i.e., the start date and end date).

(E) For work experience to be considered by the Contracting Authority the résumé must not simply indicate the title of the individual's position, but must demonstrate that the resource has the required work experience by explaining the responsibilities and work performed by the individual while in that position. The Bidder should provide complete details as to where, when, month and year, and how, through which activities/responsibilities, the stated qualifications/experience were obtained. In situations in which a proposed resource worked at the same time on more than one project, only one project will be counted toward any requirements that relate to the individual's length of experience.

(iv) The Bidder must include a proposed draft implementation plan, which demonstrates that the Bidder's proposed implementation plan meets all the mandatory requirements for implementation described in this RFP.

(v) **Customer Reference Contract Information:** The Bidder must provide customer references who must confirm, if requested by Canada, the facts identified in the Bidder's proposal. For each customer reference, the Bidder must, at a minimum, provide the name and either the telephone number or e-mail address for a contact person. Bidders are also requested to include the title of the contact person. If the named individual is unavailable when required during the evaluation period, the Bidder may provide the name and contact information of an alternate contact from the same customer.

(vi) **List of Proposed Software**: The Bidder must include a complete list identifying both the name and the version number of each component of the Licensed Software required for the proposed Software Solution.

(vii) **Solution System Architecture**: The Bidder must include an overview of the proposed Software Solution's technical architecture.

3.2B - Section II: Financial Bid

(a) **Pricing**: The total amount of Goods and Services Tax or Harmonized Sales Tax must be shown separately, if applicable. Prices must be firm prices.

(b) **All Costs to be Included:** The financial bid must include all costs for the requirement described in the bid solicitation for the entire Contract Period, including any option years. The identification of all necessary equipment, software, peripherals, cabling and components required to meet the requirements of the bid solicitation and the associated costs of these items is the sole responsibility of the Bidder.

(c) **Blank Prices**: Bidders are requested to insert "$0.00" for any item for which it does not intend to charge or for items that are already included in other prices set out in the tables. If the Bidder leaves any price blank, Canada will treat the price as "$0.00" for evaluation purposes.

3.2C - Section III: Certifications

Bidders must submit the certifications required under Part 5.

PART 4 - EVALUATION PROCEDURES AND BASIS OF SELECTION

4.1 Evaluation Procedures

(a) Bids will be assessed in accordance with the entire requirement of the bid solicitation including the technical and financial evaluation criteria. There are several steps in the evaluation process, which are described below. Even though the evaluation and selection will be conducted in steps, the fact that Canada has proceeded to a later step does not mean that Canada has conclusively determined that the Bidder has successfully passed all the previous steps. Canada may conduct steps of the evaluation in parallel.

(b) An evaluation team composed of representatives of the Client(s) and/or PWGSC will evaluate the bids on behalf of Canada. Canada may hire any independent consultant, or use any Government resources, to evaluate any bid. Not all members of the evaluation team will necessarily participate in all aspects of the evaluation.

(c) In addition to any other time periods established in the bid solicitation:

(i) **Requests for Clarifications**: If Canada seeks clarification or verification from the Bidder about its bid, the Bidder will have 2 working days (or a longer period if specified in writing by the Contracting Authority) to provide the necessary information to Canada. Failure to meet this deadline will result in the bid being declared non-responsive.

(ii) **Requests for Interviews**: If Canada wishes to interview the Bidder and/or any or all of the resources proposed by the Bidder to fulfill the requirements of the bid solicitation, the Bidder will have 2 working days (or a longer period if specified in writing by the Contracting Authority) following notice by the Contracting Authority to make any necessary arrangements (at the Bidder's sole cost) for the interview to take place at a location specified by Canada.

(iii) **Extension of Time**: If additional time is required by the Bidder, the Contracting Authority may grant an extension in his or her sole discretion.

4.2 Technical Evaluation:

(a) Mandatory Technical Criteria:

Each bid will be reviewed for compliance with the mandatory requirements of the bid solicitation. All elements of the bid solicitation that are mandatory requirements are identified specifically with the words "must" or "mandatory". Bids that do not comply with each and every mandatory requirement will be considered non-responsive and be disqualified.

The mandatory requirements are specified in the Annex D, Evaluation Criteria.

(b) Point-Rated Technical Criteria: Point-Rated Technical Criteria are specified in Annex D, Evaluation Criteria. Each bid will be rated by assigning a score to the rated requirements, which are identified in the bid solicitation by the word "rated" or by reference to a score. Bidders who fail to submit complete bids with all the information requested by this bid solicitation will be rated accordingly. Point-rated evaluation criteria and/or evaluation processes are described in the Annex D.

(c) Resource Qualifications: The qualifications and experience of the proposed resource(s) will be assessed against the requirements set out in the bid solicitation. Canada may request proof of successful completion of formal training, as well as reference information. The Contracting Authority reserves the right to request references from a Bidder to conduct a reference check to verify the accuracy of the

information provided. For each customer reference, the Bidder must provide the name, telephone number and e-mail address (unless this individual does not have an e-mail address) for a contact person. The title of each person is requested but not required. If the named individual is unavailable when required during the evaluation period, the Bidder may provide the name and contact information of an alternate contact from the same customer.

(d) **Reference Checks:** If reference checks are conducted by Canada, they will be conducted in writing by e-mail (unless the contact at the reference is only available by telephone). Canada will send all e-mail reference check requests to contacts supplied by all the Bidders on the same day. Canada will not award any points unless the response is received within 5 working days. Wherever information provided by a reference differs from the information supplied by the Bidder, the information supplied by the reference will be the information evaluated. Points will only be allocated if the reference customer is an outside client of the Bidder itself and not that of an affiliate (for example, the outside client cannot be the customer of an affiliate of the Bidder.) Points will not be allocated if the outside client is itself an affiliate or other entity that does not deal at arm's length with the Bidder. Crown references will be accepted.

(e) Consideration of Additional Software Use Terms included in Top-Ranked Bid (following financial evaluation)

(i) Acceptance of all the terms and conditions contained in Part 7 - Resulting Contract Clauses (including those relating to software licensing and those incorporated by reference) is a mandatory requirement of this bid solicitation.

(ii) However, Bidders may, as part of their bid, submit additional software use terms. Whether or not those software use terms will be included in any resulting contract (as an Annex in accordance with the Article entitled "Priority of Documents" in the Resulting Contract Clauses) will be determined using the process described below. Whether or not any proposed additional software use terms are acceptable to Canada is a matter solely within the discretion of Canada.

(iii) The process is as follows:

(A) Bids may include additional software use terms that are proposed to supplement the terms of the Resulting Contract Clauses. Bidders should not submit a software publisher's full standard license terms (because full standard license terms generally contain provisions that deal with more than simply how the software can be used; for example, they frequently deal with issues such as limitation of liability or warranty, neither of which are software use terms);

(B) In cases where the Bidder has submitted a software publisher's full standard license terms, Canada will require that the Bidder remove these terms and submit only the software use terms that the Bidder would like Canada to consider;

(C) Canada will review the additional software use terms proposed by the top-ranked Bidder (identified after the financial evaluation) to determine if there are any provisions proposed by the Bidder that are unacceptable to Canada;

(D) If Canada determines that any proposed software use term is unacceptable to Canada, Canada will notify the Bidder, in writing, and will provide the Bidder with an opportunity to remove that provision from its bid or to propose alternate language for consideration by Canada. Canada may set a time limit for the Bidder to respond; if the Bidder submits alternate language, if Canada does

not find the alternate language acceptable, Canada is not required to allow the Bidder to submit further alternate language;

(E) If the Bidder refuses to remove provisions unacceptable to Canada from its bid within the time limit set by Canada in its notice, the bid will be considered non-responsive and be disqualified; Canada may then proceed to the next-ranked bid; and

(F) If the Bidder agrees to remove the provisions that are unacceptable to Canada and it is awarded any resulting contract, the proposed additional software use terms (as revised) will be incorporated as an annex to the contract, as set out in the Article entitled "Priority of Documents" in the Resulting Contract Clauses.

(iv) For greater certainty and to ensure that only additional software use terms that have been approved by both parties are incorporated into any resulting contract, unless the additional software use terms proposed by the Bidder are included as a separate annex to the Contract and initialed by both parties, they will not be considered part of any resulting contract (even if they are part of the bid that is incorporated by reference into the resulting contract). The fact that some additional terms and conditions or software use terms were included in the bid will not result in those terms applying to any resulting contract, regardless of whether or not Canada has objected to them under the procedures described above.

(f) Technically Responsive Proposal: A technically responsive proposal is a proposal that:

i. Meets the mandatory requirements and obtains the required minimum points specified in the bid solicitation for the criteria that are subject to point rating.

4.3 Financial Evaluation

(a) Unless otherwise specified in the RFP, the financial evaluation will be conducted by calculating the Total Bid Price using the Pricing Tables completed by the bidders. The Bidder must provide a firm, all inclusive, price for the software solution and firm, all inclusive per diem rates for resource categories being proposed in accordance with the bid solicitation, which includes an initial contract period and one irrevocable one-year option period.

(b) **Mandatory Financial Criteria**

(i) **Formulae in Pricing Tables**

If the pricing tables provided to bidders include any formulae, Canada may re-input the prices provided by bidders into a fresh table, if Canada believes that the formulae may no longer be functioning properly in the version submitted by a bidder.

(ii) **Substantiation of Professional Services Rates**

In Canada's experience, bidders will from time to time propose rates at the time of bidding for one or more categories of resources that they later refuse to honour, on the basis that these rates do not allow them to recover their own costs and/or make a profit. When evaluating the rates for professional services bid, Canada may, but will have no obligation to, require price support for any rates proposed (either for all or for specific resource categories). Examples of price support that Canada would consider satisfactory include:

a. documentation (such as billing records) that shows that the Bidder has recently provided and invoiced another customer (with whom the Bidder deals at arm's length) for services similar to the services that would be provided by the relevant resource category, where those services were provided for at least one month and the fees charged are equal to or less than the rate offered to Canada (to protect the privacy of the customer, the Bidder may black out the customer's name and personal information on the invoice submitted to Canada);

b. a signed contract between the Bidder and an individual qualified (based on the qualifications described in this bid solicitation) to provide services under the relevant resource category, where the amount payable under that contract by the Bidder to the resource is equal to or less than the rate bid for that resource category;

c. a signed contract with a subcontractor who will perform the work under any resulting contract, which provides that the required services will be provided at a rate that is equal to or less than the rate bid for the relevant resource category (and where the resource meets all the qualifications described in this bid solicitation); or

d. details regarding the salary paid to and benefits provided to the individuals employed by the Bidder qualified (based on the qualifications described in this bid solicitation) to provide services under the relevant resource category, where the amount of compensation, when converted to a per diem or hourly rate (as applicable), is equal to or less than the rate bid for that resource category.

Once Canada requests substantiation of the rates bid for any resource category, it is the sole responsibility of the Bidder to submit information (either the information described in the examples above, or other information that demonstrates that it will be able to recover its own costs based on the rates it has proposed) that will allow Canada to determine whether it can rely, with confidence, on the Bidder's ability to provide the required services at the rates bid, while, at a minimum, recovering its own costs. Where Canada determines that the information provided by the Bidder does not demonstrate the Bidder's ability to recover its own costs in providing the relevant resource, Canada may declare the bid non-compliant, if the rate bid is 20% of or lower than the median price bid by compliant bidders for the first year of the resulting contract for the relevant resource(s). Only the Firm Per Diem Rates of proposals that are technically responsive will be considered.

4.4 Basis of Selection

(a) A bid must comply with the requirements of the bid solicitation and meet all mandatory criteria to be declared responsive. The responsive bid that obtains the highest combined rating of technical merit and price, by adding the technical score with the financial score, will be recommended for award of a contract. The total possible technical score is 65 while the total possible financial score is 35.

Calculation of Composite Score

The Composite Score will be determined as follows:

Technical portion: 65%

Price portion: 35%.

Total of technical points (maximum 65 points) = Number of technical points.

Lowest total price quoted, divided by the total price quoted, multiplied by 35 = Number of price points.

Number of technical points + Number of price points = Composite Score.

Example:

Bidder	Bidder 1	Bidder 2	Bidder 3
Technical Points	58	52	46
Total Bid Price	60,000	55,000	*50,000
Calculation	Technical Points	Price Points	Total Points
Bidder 1	58	*50/60 X 35 = 29.17	87.17
Bidder 2	52	50/55 X 35 = 31.82	83.82
Bidder 3	46	50/50 X 35 = 35.00	81.00

** Represents lowest bid price*

(b) Bidders should note that all contract awards are subject to Canada's internal approvals process, which includes a requirement to approve funding in the amount of any proposed contract. Despite the fact that the Bidder may have been recommended for contract award, a contract will only be awarded if internal approval is granted according to Canada's internal policies. If approval is not granted, no contract will be awarded.

(c) If more than one bidder is ranked first because of identical overall scores, then the bidder with the best financial score will become the top-ranked bidder.

(d) Notification of Evaluation Results:

All SA Holders who respond to a SBIPS RFP will be notified in writing regarding the outcome of the RFP process. This notice will include the following information:

i. Solicitation Number;
ii. Company name of winning bidder including total points scored (for multiple resource requirements only);
iii. Total value of contract awarded;
iv. Number of responses received by the Contracting Authority; and
v. Total points scored per individual bidder (Note: bidders will only receive their own total points scored and not the score of the other bidders)

PART 5 – CERTIFICATIONS

Bidders must provide the required certifications to be awarded a contract. Canada will declare a bid non-responsive if the required certifications are not completed and submitted in accordance with the articles below.

Compliance with the certifications bidders provide to Canada is subject to verification by Canada during the bid evaluation period (before award of a contract) and after award of a contract. The Contracting Authority will have the right to ask for additional information to verify bidders' compliance with the certifications before award of a contract. The bid will be declared non-responsive if any certification made by the Bidder is untrue, whether made knowingly or unknowingly. Failure to comply with the certifications or to comply with the request of the Contracting Authority for additional information will also render the bid non-responsive.

Certification is Required with the Bid at Bid Closing: Bidders must submit, at bid closing, the duly completed certification as part of their bid.

5.1 Federal Contractors Program for Employment Equity - Certification

(a) The Federal Contractors Program (FCP) requires that some suppliers, including a supplier who is a member of a joint venture, bidding for federal government contracts, valued at $200,000 or more (including all applicable taxes), make a formal commitment to implement employment equity. This is a condition precedent to contract award. If the Bidder, or, if the Bidder is a joint venture and if any member of the joint venture, is subject to the FCP, evidence of its commitment must be provided before the award of the Contract.

(b) Suppliers who have been declared ineligible contractors by Human Resources and Skills Development Canada (HRSDC) are no longer eligible to receive government contracts over the threshold for solicitation of bids as set out in the Government Contracts Regulations. Suppliers may be declared ineligible contractors either as a result of a finding of non-compliance by HRSDC, or following their voluntary withdrawal from the FCP for a reason other than the reduction of their workforce to less than 100 employees.
Any bids from ineligible contractors, including a bid from a joint venture that has a member who is an ineligible contractor, will be declared non-responsive.

(c) If the Bidder does not fall within the exceptions enumerated in 3.(a) or (b) below, or does not have a valid certificate number confirming its adherence to the FCP, the Bidder must fax (819-953-8768) a copy of the signed form LAB 1168, Certificate of Commitment to Implement Employment Equity, to the Labour Branch of HRSDC.

(d) The Bidder, or, if the Bidder is a joint venture the member of the joint venture, certifies its status with the FCP, as follows:

The Bidder or the member of the joint venture

(i) ( ) is not subject to the FCP, having a workforce of less than 100 full-time or part-time permanent employees, or temporary employees having worked 12 weeks or more in Canada;

(ii) ( ) is not subject to the FCP, being a regulated employer under the Employment Equity Act, S.C. 1995, c. 44;

(iii) ( ) is subject to the requirements of the FCP, having a workforce of 100 or more full-time or part-time permanent employees, or temporary employees having worked 12 weeks or more in Canada, but has not previously obtained a certificate number from HRSDC (having not bid on

requirements of $200,000 or more), in which case a duly signed certificate of commitment is attached;

(iv) ( ) is subject to the FCP, and has a valid certificate number as follows: _____ (e.g. has not been declared an ineligible contractor by HRSDC.)

Further information on the FCP is available on the HRSDC Web site.

Note to Bidders: *Bidders are requested to use the Bid Submission Form to provide information about their status under this program. For a joint venture bidder, this information must be provided for each member of the joint venture.*

5.2 Former Public Servant Certification

Standard Acquisition Clause and Conditions Clause A3025T (2012-11-19) Former Public Servant Certification is hereby incorporated by reference.

Note to Bidders: *Bidders are requested to provide the information required by this clause in their Bid Submission Form.*

5.3 Status and Availability of Resources

(a) By submitting a bid, the Bidder certifies that, should it be awarded a contract as a result of the bid solicitation, every individual proposed in its bid will be available to perform the Work as required by Canada's representatives and at the time specified in the bid solicitation or agreed to with Canada's representatives.

(b) If for reasons beyond its control, the Bidder is unable to provide the services of an individual named in its bid, the Bidder may propose a substitute with similar qualifications and experience. The Bidder must advise the Contracting Authority of the reason for the substitution and provide the name, qualifications and experience of the proposed replacement. For the purposes of this clause, only the following reasons will be considered as beyond the control of the Bidder: death, sickness, retirement, resignation, dismissal for cause or termination of an agreement for default.

(c) If the Bidder has proposed any individual who is not an employee of the Bidder, by submitting a bid, the Bidder certifies that it has the permission from that individual to propose his/her services in relation to the Work to be performed and to submit his/her résumé to Canada. The Bidder must, upon request from the Contracting Authority, provide a written confirmation, signed by the individual, of the permission given to the Bidder and of his/her availability. Failure to comply with the request may result in the bid being declared non-responsive.

5.4 Education and Experience

(a) SACC Manual Clause A3010T (2010-08-16) Education and Experience

PART 6 - SECURITY, FINANCIAL AND OTHER REQUIREMENTS

6.1 Mandatory at Bid Closing - Security Requirement

(a) At the date of bid closing, the following conditions must be met:

(i) The Bidder must hold a valid organization security clearance as indicated in Part 7 - Resulting Contract Clauses;

(ii) the Bidder's proposed individuals requiring access to classified or protected information, assets or sensitive work site(s) must each meet the security requirement as indicated in Part 7 - Resulting Contract Clauses; and

(iii) the Bidder must provide the name of all individuals who will require access to classified or protected information, assets or sensitive work sites as follows:

(1) Name of individual as it appears on security clearance application;

(2) Level of security clearance obtained and expiry date;

(3) Security screening Certificate and Briefing Form file number.

(b) Additional information on security requirements, bidders should consult the "Security Requirements for PWGSC Bid Solicitations - Instructions to Bidders" document (http://www.tpsgc-pwgsc.gc.ca/app-acq/lc-pl/lc-pl-eng.html#a31) on the Departmental Standard Procurement Documents Website.

(c) In the case of a joint venture bidder, each member of the joint venture must meet the security requirements.

6.2 Financial Capability

(a) SACC Manual clause A9033T (2012-07-16) Financial Capability applies to this requirement.

(b) In the case of a joint venture bidder, each member of the joint venture must meet the financial capability requirements.

PART 7 - RESULTING CONTRACT CLAUSES

The following clauses apply to and form part of any contract resulting from the bid solicitation.

7.1 Requirement

_____ (the Contractor) agrees to supply to the Client the goods and services described in the Contract, including the Statement of Work, in accordance with and at the prices set out in the Contract. This includes:

(i) providing professional services, as requested by Canada;

(ii) providing the simulation software application described in the Contract;

(iii) providing the Software Documentation;

(a) **Client(s):** Any reference to "**Client(s)**" includes any Government Department, Departmental Corporation or Agency, or other Crown entity described in the *Financial Administration Act* (as amended from time to time), and any other party for which the Department of Public Works and Government Services has been authorized to act from time to time under section 16 of the *Department of Public Works and Government Services Act*.

(b) **Reorganization of Client**: The Contractor's obligation to perform the Work will not be affected by (and no additional fees will be payable as a result of) the renaming, reorganization, reconfiguration, or restructuring of any Client. The reorganization, reconfiguration and restructuring of the Client includes the privatization of the Client, its merger with another entity, or its dissolution, where that dissolution is followed by the creation of another entity or entities with mandates similar to the original Client.

(c) **Defined Terms**: Words and expressions defined in the General Conditions or Supplemental General Conditions and used in the Contract have the meanings given to them in the General Conditions or Supplemental General Conditions.

(i) any reference to a "**deliverable**" or "**deliverables**" includes the software, the license to use the Licensed Software, the source code and documentations.

(d) **Location of Services:** Services must be delivered as requested to the locations specified in the Contract, which delivery locations must exclude any area subject to one of the Comprehensive Land Claim Agreements (CLCAs).

7.2 Optional Goods and/or Services

(a) The Contractor grants to Canada the irrevocable option to acquire the services described at the Contract under the same terms and conditions and at the prices and/or rates stated in the Contract. The option may only be exercised by the Contracting Authority by notice in writing and will be evidenced, for administrative purposes only, through a contract amendment.

(b) The Contracting Authority may exercise the option at any time before the expiry of the Contract by sending a written notice to the Contractor.

7.3 Task Authorization

(a) **Purpose of TA**: Services to be provided under the Contract on an as-and-when-requested basis will be ordered by Canada using the Task Authorization ("TA") process.

(b) **Process of Issuing a TA**: If a requirement for a specific task is identified, a draft "statement of task" will be prepared by the Technical Authority and

sent to the Contractor. Once it receives the statement of task, the Contractor must submit a response to the Authority identified in the TA detailing the cost and time to complete the task. The Contractor's response must be based on the rates set out in the Contract. The Contractor will not be paid for providing the response or for providing other information required to prepare and issue the TA. The Contractor must provide any information requested by Canada in relation to the preparation of a TA within three working days of the request, unless otherwise specified.

(c) **Approval Process**: If Canada approves the Contractor's task response, Canada (by its authorized representative, as described in this contract) will issue the TA by forwarding a signed copy of the final TA form to the Contractor. Whether or not to approve or issue a TA is entirely within Canada's discretion.

(d) **Authority to Issue a TA**: Any TA with a value less than or equal to $200,000.00 (including GST/HST) may be issued by the Technical Authority. Any TA with a value greater than this amount must be issued directly by the Contracting Authority. By providing written notice to the Contractor, the Contracting Authority may suspend the Project Authority's authority to issue TAs at any time.

(e) **Contents of a TA**: A Task Authorization must contain the following information, if applicable:

(i) a task number;

(ii) the details of any financial coding to be used;

(iii) the number of resources in each category required;

(iv) a statement of work for the task outlining the activities to be performed and identifying any deliverables;

(v) the duration of the task to be carried out (start and end dates);

(vi) milestone dates for deliverables and payments (if applicable);

(vii) the number of person-days of effort required;

(viii) the specific work location;

(ix) the price payable to the Contractor for performing the task, with an indication of whether it is a firm price or a maximum TA price (and, for maximum price task authorizations, the TA must indicate how the final amount payable will be determined; where the TA does not indicate how the final amount payable will be determined, the amount payable is the amount, up to the maximum, that the Contractor demonstrates was actually worked on the project, by submitting time sheets filled in at the time of the work by the individual resources to support the charges); and

(x) any other constraints that might affect the completion of the task.

(f) **Charges for Work under a TA**: The Contractor must not charge Canada anything more than the price set out in the Task Authorization unless Canada has issued a TA amendment authorizing the increased expenditure. Canada will not pay the Contractor for any design changes, modifications or interpretations of the Work unless they have been approved, in writing, by the Contracting Authority before being incorporated into the Work.

(g) **Consolidation of TAs for Administrative Purposes**: The Contract may be amended from time to time to reflect all Task Authorizations issued and approved by the Contracting Authority to date, to document the Work performed under those TAs for administrative purposes.

(h) **Refusal of Task Authorizations**:

Unless specified otherwise in the RFP, the Contractor is required to submit a response to every draft statement of task issued by Canada. Canada may immediately, and without further notice, terminate the Contract for default if the Contractor does not submit a response to a draft statement of task issued during the Contract Period.

7.4 Minimum Work Guarantee

(a) In this clause, "**Minimum Contract Value**" means 5% of the amount identified as the Total Estimated Cost on page 1 of the Contract when it is first awarded.

(b) The Contractor must perform the Work described in the Contract as and when requested by Canada during the Contract Period. Canada's obligation under the Contract is to request Work in the amount of the Minimum Contract Value or, at Canada's option, to pay the Contractor at the end of the Contract in accordance with paragraph (c). In consideration of this obligation, the Contractor agrees to stand in readiness throughout the Contract Period to perform the Work described in the Contract.

(c) If Canada does not request work in the amount of the Minimum Contract Value during the Contract Period, Canada must pay the Contractor the difference between the Minimum Contract Value and the cost of the Work performed.

(d) Canada will have no obligation to the Contractor under this clause if Canada terminates the Contract in whole or in part for default.

7.5 Standard Clauses and Conditions

All clauses and conditions identified in the Contract by number, date and title are set out in the Standard Acquisition Clauses and Conditions Manual (http://sacc.pwgsc.gc.ca/sacc/index-e.jsp) issued by Public Works and Government Services Canada.

(a) **General Conditions:**

2035 (2012-11-19), General Conditions - Higher Complexity - Services, apply to and form part of the Contract.

(b) **Supplemental General Conditions :**

The following Supplemental General Conditions:

(i) 4002 (2010-08-16), Supplemental General Conditions - Software Development or Modification Services;

(ii) 4007 (2010-08-16), Supplemental General Conditions - Canada to Own Intellectual Property Rights in Foreground Information;

apply to and form part of the Contract.

(c) **Modifications to Supplemental General Conditions :**

(i) Subsection 8.3 of 4002 - Supplemental General Conditions – Software Development or Modification Services is amended as follows:

- Delete sub article (b).

(ii) Subsection 4.2 of 4007 - Supplemental General Conditions - Canada to Own Intellectual Property Rights in Foreground Information is amended as follows:

- Insert the following sub article e., "the right to disclose, distribute, sublicense and sell the Background Information to third parties".

(iii) Subsection 4.3 of 4007 - Supplemental General Conditions - Canada to Own Intellectual Property Rights in Foreground Information is amended as follows:

- Delete the following sentence: "Furthermore, in case of commercial off-the-shelf software, the contractor's obligation to make the source code promptly available to Canada applies only to source code that is within the control or can be obtained by the Contractor or any subcontractor".

7.6 Security Requirement

1. The Contractor must at all times during the performance of the Contract/SA, hold a valid Designated Organization Screening (DOS), issued by the Canadian Industrial Security Directorate (CISD), Public Works and Government Services Canada (PWGSC).
2. The Contractor personnel requiring access to PROTECTED A, B, or C information as required, assets or sensitive work site(s) must EACH hold a valid RELIABILITY STATUS, granted or approved by CISD/PWGSC.
3. The Contractor MUST NOT remove any PROTECTED information or assets from the identified work site(s), and the Contractor must ensure that its personnel are made aware of and comply with this restriction.
4. Subcontracts which contain security requirements are NOT to be awarded without the prior written permission of CIISD/PWGSC.
5. The Contractor must comply with the provisions of the:
    1. Security Requirements Check List EN537-05IT01-SBIPSG1 **and security guide (if applicable)**, described in Annex "A";
    2. Industrial Security Manual (Latest Edition).

7.7 Contract Period

(a) Contract Period: The "**Contract Period**" is the entire period of time during which the Contractor is obliged to perform the Work, which includes:

(i) The "**Initial Contract Period**", which begins on the date the Contract is awarded and ends 2 and half years later; and

(ii) The period during which the Contract is extended, if Canada chooses to exercise the option set out in the Contract.

(b) Option to Extend the Contract:

(i) The Contractor grants to Canada the irrevocable option to extend the term of the Contract by 1 additional one-year period under the same terms and conditions. The Contractor agrees that, during the extended period of the Contract, it will be paid in accordance with the applicable provisions set out in the Basis of Payment.

(ii) Canada may exercise this option at any time by sending a written notice to the Contractor at least 30 calendar days before the expiry date of the Contract. The option may only be exercised by the Contracting Authority, and will be evidenced, for administrative purposes only, through a formal contract amendment.

7.8 Authorities

(a) Contracting Authority

The Contracting Authority for the Contract is:

Contracting Authority:	Anoule Pierre
E-mail Address:	anoule.pierre@tpsgc.gc.ca
Telephone:	(819) 956-2137
Facsimile:	(819) 956-1207

The Contracting Authority is responsible for the management of the Contract and any changes to the Contract must be authorized in writing by the Contracting Authority. The Contractor must not perform work in excess of or outside the scope of the Contract based on verbal or written requests or instructions from anybody other than the Contracting Authority.

(b) Technical Authority (To be provided at time of Contract award)

The Technical Authority for the Contract is:

Name: _____
Title: _____
Organization: _____
Address: _____
Telephone: _____
Facsimile: _____
E-mail address: _____

The Technical Authority is the representative of the department or agency for whom the Work is being carried out under the Contract and is responsible for all matters concerning the technical content of the Work under the Contract. Technical matters may be discussed with the Technical Authority; however, the Technical Authority has no authority to authorize changes to the scope of the Work. Changes to the scope of the Work can only be made through a contract amendment issued by the Contracting Authority.

(c) Contractor's Representative (To be provided at time of Contract award)

7.9 Payment

(a) Basis of Payment

(i) **Software Application provided with a Firm Price**: For professional services requested by Canada, Canada will pay the Contractor the firm price set out in the Contract (based on the firm price set out in Annex B of this contract), GST/HST extra

Firm Price: $_____

(ii) **Professional Services provided under a Task Authorization with a Maximum Price**: For professional services requested by Canada, in accordance with an approved Task Authorization, Canada will pay the Contractor the firm/maximum price set out in the Task Authorization (based on the firm, all-inclusive per diem rates set out in Annex B of this contract), GST/HST extra

Estimated Cost: $_____

(iii) GST/HST: Estimated Cost: $_____

(iv) **Competitive Award**: The Contractor acknowledges that the Contract has been awarded as a result of a competitive process. No additional charges will be allowed to compensate for errors, oversights, misconceptions or underestimates made by the Contractor when bidding for the Contract.

(v) **Professional Services Rates**: In Canada's experience, bidders from time to time propose rates at the time of bidding for one or more categories of resources that they later refuse to honour, on the basis that these rates do not allow them to recover their own costs and/or make a profit. This denies Canada of the benefit of the awarded contract. If the Contractor refuses, or is unable, to provide an individual with the qualifications described in the Contract within the time described in the Contract (or proposes instead to provide someone from an alternate category at a different rate), whether or not Canada terminates the Contract as a whole, Canada may impose sanctions or take other measures in accordance with the PWGSC Vendor Performance Policy (or equivalent) then in effect, which may include prohibiting the Contractor from bidding on future requirements that include any professional services, or rejecting the Contractor's other bids for professional services requirements on the basis that the Contractor's performance on this or other contracts is sufficiently poor to jeopardize the successful completion of other requirements.

(vi) **Purpose of Estimates**: All estimated costs contained in the Contract are included solely for the administrative purposes of Canada and do not represent a commitment on the part of Canada to purchase services in these amounts. Any commitment to purchase specific amounts or values of services are described elsewhere in the Contract.

(b) Limitation of Expenditure

Canada will not pay the Contractor for any design changes, modifications or interpretations of the Work unless they have been approved, in writing, by the Contracting Authority before their incorporation into the Work.

(c) Method of Payment

1. Software Application, Milestone Payment

Canada will pay the Contractor upon completion and delivery of the Work for each phase as set out in the annex "B" Basis of Payment, in accordance with the payment provisions of the Contract if:

(i) an accurate and complete invoice and any other documents required by the Contract have been submitted in accordance with the invoicing instructions provided in the Contract;
(ii) all such documents have been verified by Canada;
(iii) the Work delivered has been accepted by Canada.

2. Task Authorizations with a Maximum Price or- Milestone Payments

(i) For any Task Authorization/Contract issued that includes a schedule of milestone payments to be made once specific portions of the work have been completed and accepted, Canada will make milestone payments in accordance with the schedule of milestones detailed in that Task Authorization and the payment provisions of the Contract, up to 50% of the amount claimed and approved by Canada if:

(A) an accurate and complete claim for milestone payment using form PWGSC-TPSGC 1111 http://publiservice.tpsgc-pwgsc.gc.ca/forms/text/index-e.html and any other documents required by the Contract have been submitted in accordance with the invoicing instructions provided in the Contract.

(B) the total amount for all milestone payments paid by Canada under that Task Authorization/Contract does not exceed 90% percent, or the other percentage specified in the Task Authorization/Contract, of the total amount to be paid;

(C) all the certificates appearing on form PWGSC-TPSGC 1111 have been signed by the respective authorized representatives; and

(D) all work associated with the milestone and any deliverable required have been completed, delivered, and accepted by Canada.

(ii) The balance of the amount payable will be paid in accordance with the basis of payment provisions of the Contract following delivery and acceptance of the Work for which milestone payments were made.

(d) Time Verification

Time charged and the accuracy of the Contractor's time recording system are subject to verification by Canada, before or after payment is made to the Contractor. If verification is done after payment, the Contract must repay any overpayment, at Canada's request.

7.10 Invoicing Instructions

(a) The Contractor must submit invoices in accordance with the information required in the General Conditions.

(b) The Contractor's invoice must include a separate line item for each subparagraph in the Basis of Payment provision.

(c) By submitting invoices (other than for any items subject to an advance payment), the Contractor is certifying that the goods and services have been delivered and that all charges are in accordance with the Basis of Payment provision of the Contract, including any charges for work performed by subcontractors.

(d) The Contractor must provide the original of each invoice to the Technical Authority, and a copy to the Contracting Authority.

7.11 Certifications

Compliance with the certifications provided by the Contractor in its response to the RFP is a condition of the Contract and subject to verification by Canada during the entire Contract Period. If the Contractor does not comply with any certification or it is determined that any certification made by the Contractor in its bid is untrue, whether made knowingly or unknowingly, Canada has the right, under the default provision of the Contract, to terminate the Contract for default.

7.12 Applicable Laws

The Contract must be interpreted and governed, and the relations between the parties determined, by the laws in force in the province of Ontario or as indicated in the Bidder's Supply Arrangement.

7.13 Priority of Documents

If there is a discrepancy between the wording of any documents that appear on the following list, the wording of the document that first appears on the list has priority over the wording of any document that appears later on the list:

(a) these Articles of Agreement, including any individual SACC clauses incorporated by reference in these Articles of Agreement;

(b) Supplemental general conditions, in the following order:

(i) 4002 (2010-08-16), Supplemental General Conditions - Software Development or Modification Services;

(ii) 4007 (2010-08-16), Supplemental General Conditions - Canada to Own Intellectual Property Rights in Foreground Information;

(c) General Conditions 2035 (2012-11-19);

(d) Statement of Work;

(e) Basis of Payment;

(f) Security Requirements Check List;

(g) the signed Task Authorizations;

(h) Additional Software Use Terms Approved by Canada (if any), which are only binding on Canada if they have been initialed by both parties in the signed copy of the Contract;

(i) Supply Arrangement EN537-05IT01/XXX/EI

(j) the Contractor's bid dated _____ (**insert date of bid**), as amended _____ (**insert date(s) of amendment(s) if applicable**), not including any software publisher license terms and conditions that may be included in the bid, not including any provisions in the bid with respect to limitations on liability, and not including any terms and conditions incorporated by reference (including by way of a web link) in the bid.

7.14 Foreign Nationals (Canadian Contractor)

SACC Manual clause A2000C (2006-06-16) Foreign Nationals (Canadian Contractor)

Note to Bidders: *Either this clause or the one that follows, whichever applies (based on whether the successful bidder is a Canadian Contractor or Foreign Contractor), will be included in any resulting contract.*

7.15 Foreign Nationals (Foreign Contractor)

SACC Manual clause A2001C (2006-06-16) Foreign Nationals (Foreign Contractor)

7.16 Insurance Requirements

The Contractor is responsible for deciding if insurance coverage is necessary to fulfill its obligation under the Contract and to ensure compliance with any applicable law. Any insurance acquired or maintained by the Contractor is at its own expense and for its own benefit and protection. It does not release the Contractor from or reduce its liability under the Contract.

7.17 Limitation of Liability - Information Management/Information Technology

1. This section applies despite any other provision of the Contract and replaces the section of the general conditions entitled "Liability". Any reference in this section to damages caused by the Contractor also includes damages caused by its employees, as well as its subcontractors, agents, and representatives, and any of their employees. This section applies regardless of whether the claim is based in contract, tort, or another cause of action. The Contractor is not liable to Canada with respect to the performance of or failure to perform the Contract, except as described in this section and in any section of the Contract pre-establishing any liquidated damages. The Contractor is only liable for indirect, special or consequential damages to the extent described in this section, even if it has been made aware of the potential for those damages.

2. First Party Liability:

   (a) The Contractor is fully liable for all damages to Canada, including indirect, special or consequential damages, caused by the Contractor's performance or failure to perform the Contract that relate to:

      (i) any infringement of intellectual property rights to the extent the Contractor breaches the section of the general conditions entitled "Intellectual Property Infringement and Royalties";

      (ii) physical injury, including death.

   (b) The Contractor is liable for all direct damages caused by the Contractor's performance or failure to perform the Contract affecting real or tangible personal property owned, possessed, or occupied by Canada.

   (c) Each of the Parties is liable for all direct damages resulting from its breach of confidentiality under the Contract. Each of the Parties is also liable for all indirect, special or consequential damages in respect of its unauthorized disclosure of the other Party's trade secrets (or trade secrets of a third party provided by one Party to another under the Contract) relating to information technology.

   (d) The Contractor is liable for all direct damages relating to any encumbrance or claim relating to any portion of the Work for which Canada has made any payment. This does not apply to encumbrances or claims relating to intellectual property rights, which are addressed under (a) above.

   (e) The Contractor is also liable for any other direct damages to Canada caused by the Contractor in any way relating to the Contract, including:

      (A) any breach of the warranty obligations under the Contract, up to the total amount paid by Canada (including any applicable taxes) for the goods and services affected by the breach of warranty; and

      (B) any other direct damages, including all identifiable direct costs to Canada associated with re-procuring the Work from another party if the Contract is terminated either in whole or in part for default, up to an aggregate maximum for this subparagraph (B) of the greater of 0.75 times the total estimated cost (meaning the dollar amount shown on the first page of the Contract in the cell

titled "Total Estimated Cost" or shown on each call-up, purchase order or other document used to order goods or services under this instrument), or $1,000,000.00

In any case, the total liability of the Contractor under subparagraph (b) will not exceed the total estimated cost (as defined above) for the Contract or $1,000,000.00, whichever is more.

(f) If Canada's records or data are harmed as a result of the Contractor's negligence or willful act, the Contractor's only liability is, at the Contractor's own expense, to restore Canada's records and data using the most recent back-up kept by Canada. Canada is responsible for maintaining an adequate back-up of its records and data.

3. Third Party Claims:

(a) Regardless of whether a third party makes its claim against Canada or the Contractor, each Party agrees that it is liable for any damages that it causes to any third party in connection with the Contract as set out in a settlement agreement or as finally determined by a court of competent jurisdiction, where the court determines that the Parties are jointly and severally liable or that one Party is solely and directly liable to the third party. The amount of the liability will be the amount set out in the settlement agreement or determined by the court to have been the Party's portion of the damages to the third party. No settlement agreement is binding on a Party unless its authorized representative has approved the agreement in writing.

(b) If Canada is required, as a result of joint and several liability, to pay a third party in respect of damages caused by the Contractor, the Contractor must reimburse Canada by the amount finally determined by a court of competent jurisdiction to be the Contractor's portion of the damages to the third party. However, despite paragraph (a), with respect to special, indirect, and consequential damages of third parties covered by this section, the Contractor is only liable for reimbursing Canada for the Contractor's portion of those damages that Canada is required by a court to pay to a third party as a result of joint and several liability that relate to the infringement of a third party's intellectual property rights; physical injury of a third party, including death; damages affecting a third party's real or tangible personal property; liens or encumbrances on any portion of the Work; or breach of confidentiality.

(c) The Parties are only liable to one another for damages to third parties to the extent described in this paragraph 3.

7.18 Joint Venture Contractor

(a) The Contractor confirms that the name of the joint venture is _____ and that it is comprised of the following members: **[all the joint venture members named in the Contractor's original bid will be listed]**.

(b) With respect to the relationship among the members of the joint venture Contractor, each member agrees, represents and warrants (as applicable) that:

(i) _____ has been appointed as the "representative member" of the joint venture Contractor and has fully authority to act as agent for each member regarding all matters relating to the Contract;

(ii) by giving notice to the representative member, Canada will be considered to have given notice to all the members of the joint venture Contractor; and

(iii) all payments made by Canada to the representative member will act as a release by all the members.

(c) All the members agree that Canada may terminate the Contract in its discretion if there is a dispute among the members that, in Canada's opinion, affects the performance of the Work in any way.

(d) All the members are jointly and severally or solitarily liable for the performance of the entire Contract.

(e) The Contractor acknowledges that any change in the membership of the joint venture (i.e., a change in the number of members or the substitution of another legal entity for an existing member) constitutes an assignment and is subject to the assignment provisions of the General Conditions.

(f) The Contractor acknowledges that all security and controlled goods requirements in the Contract, if any, apply to each member of the joint venture Contractor.

Note to Bidders: This Article will be deleted if the bidder awarded the contract is not a joint venture. If the contractor is a joint venture, this clause will be completed with information provided in its bid.

7.19 Professional Services - General

(a) The Contractor must provide professional services on request as specified in this contract. Where in the Contract a specific individual is identified as required to perform the Work, the Contractor must make such person available to perform the work withing 10 working days of the issuance of the Contract or the TA (whichever first contains instructions from Canada for that individual to report to the Work site). Where such a specific individual is unavailable to perform the Work, Canada may elect to either (i) exercise its rights or remedies under the Contract or at law (including terminating the Contract for default), or (ii) Canada may require the Contractor to propose the replacement of the specific individual in accordance with the Article titled, "Replacement of Specific Individuals" in the General Conditions 2035. This obligation applies despite any changes that Canada may have made to any hardware, software or any other aspect of the Client's operating environment.

(b) If there must be a change in a resource performing work under the Contract (which must in any case comply with the requirements in the section of the General Conditions entitled "Replacement of Specific Individuals"), the Contractor must make the replacement available for work within 10 working days of the departure of the existing resource (or, if Canada has requested the replacement, within 15 working days of Canada's notice of the requirement for a replacement).

(c) All resources provided by the Contractor must meet the qualifications described in the Contract (including those relating to previous experience, professional designation, education, and language proficiency) and must be competent to provide the required services by any delivery dates described in the Contract. The resource must be approved by Canada prior to the replacement at the Work site.

(d) The Contractor must monitor its employees to ensure satisfactory performance and that progress of the Work is maintained to Canada's satisfaction. A Contractor representative will meet with the Project and/or Technical Authority on a regular basis (as specified by Canada) to discuss the performance of its resources and to resolve any issues at hand.

(e) If the Contractor fails to meet any of its obligations under this Article, or fails to deliver any deliverable or complete any task described in the Contract on time, in addition to any other rights or remedies available to Canada under the Contract or the law, Canada may notify the Contractor of the deficiency, in which case the Contractor must submit a written plan to the Technical Authority within 10 working days detailing the actions that the Contractor will undertake to remedy the deficiency. The Contractor must prepare and implement the plan at its own expense.

7.20 Safeguarding Electronic Media

(a) Before using them on Canada's equipment or sending them to Canada, the Contractor must use a regularly updated product to scan electronically all electronic media used to perform the Work for computer viruses and other coding intended to cause malfunctions. The Contractor must notify Canada if any electronic media used for the Work are found to contain computer viruses or other coding intended to cause malfunctions.

(b) If magnetically recorded information or documentation is damaged or lost while in the Contractor's care or at any time before it is delivered to Canada in accordance with the Contract, including accidental erasure, the Contractor must immediately replace it at its own expense.

7.21 Representations and Warranties

The Contractor made statements regarding its and its proposed resources experience and expertise in its bid that resulted in the award of the Contract. The Contractor represents and warrants that all those statements are true and acknowledges that Canada relied on those statements in awarding the Contract. The Contractor also represents and warrants that it has, and all its resources and subcontractors that perform the Work have, and at all times during the Contract Period they will have, the skills, qualifications, expertise and experience necessary to perform and manage the Work in accordance with the Contract, and that the Contractor (and any resources or subcontractors it uses) has previously performed similar services for other customers.

7.22 Access to Canada's Property and Facilities

Canada's property, facilities, equipment, documentation, and personnel are not automatically available to the Contractor. If the Contractor would like access to any of these, it is responsible for making a request to the Technical Authority. Unless expressly stated in the Contract, Canada has no obligation to provide any of these to the Contractor. If Canada chooses, in its discretion, to make its property, facilities, equipment, documentation or personnel available to the Contractor to perform the Work, Canada may require an adjustment to the Basis of Payment and additional security requirements may apply.

7.23 Government Property

(i) Canada agrees to supply the Contractor with the items listed below (the "**Government Property**"):

- Electronic model files and Electronic scenario data;
- The prototype software.

The section of the General Conditions entitled "Government Property" also applies to the use of the Government Property by the Contractor.

BIDDER FORMS

BID SUBMISSION FORM		
Bidder's full legal name		
Authorized Representative of Bidder for evaluation purposes (e.g., clarifications)	Name	
	Title	
	Address	
	Telephone #	
	Fax #	
	Email	
Bidder's Procurement Business Number (PBN) [*see the Standard Instructions 2003*]		
Jurisdiction of Contract: Province in Canada the bidder wishes to be the legal jurisdiction applicable to any resulting contract (if other than as specified in solicitation)		
Former Public Servants See the Article in Part 5 of the bid solicitation entitled Former Public Servant Certification for a definition of "Former Public Servant".	Is the Bidder a FPS in receipt of a pension as defined in the bid solicitation? Yes ____ No ____ If yes, provide the information required by the Article in Part 5 entitled "Former Public Servant Certification"	
	Is the Bidder a FPS who received a lump sum payment under the terms of a work force reduction program? Yes ____ No ____ If yes, provide the information required by the Article in Part 5 entitled "Former Public Servant Certification"	

Canadian Content Certification As described in the solicitation, bids with at least 80% Canadian content are being given a preference. [For the definition of Canadian goods and services, consult the PWGSC SACC clause A3050T]	On behalf of the bidder, by signing below, I confirm that [*check the box that applies*]:	
	At least 80 percent of the bid price consists of Canadian goods and services (as defined in the solicitation)	
	Less than 80 percent of the bid price consists of Canadian goods and services (as defined in the solicitation)	
Federal Contractors Program for Employment Equity (FCP EE) Certification: If the bidder is exempt, please indicate the basis for the exemption to the right. If the bidder does not fall within the exceptions enumerated to the right, the Program requirements do apply and the bidder is required either to: (a)submit to the Department of HRSD form LAB 1168, Certificate of Commitment to Implement Employment Equity, DULY SIGNED; or (b)submit a valid Certificate number confirming its adherence to the FCP-EE. Bidders are requested to include their FCP EE Certification or signed LAB 1168 with their bid; if this information is not provided in the bid, it must be provided upon request by the Contracting Authority during evaluation. For a joint venture bidder, this information must be provided for each member of the joint venture.	On behalf of the bidder, by signing below, I also confirm that the bidder [*check the box that applies*]:	
	(a) is not subject to Federal Contractors Program for Employment Equity (FCP-EE), because it has a workforce of less than 100 permanent full or part-time employees in Canada;	

	(b) is not subject to FCP-EE, because it is a regulated employer under the *Employment Equity Act*;	
	(c) is subject to the requirements of FCP-EE, because it has a workforce of 100 or more permanent full or part-time employees in Canada, but has not previously obtained a certificate number from the Department of Human Resources and Skills Development (HRSD) (having not bid on requirements of $200,000 or more), in which case a duly signed certificate of commitment is attached; OR	
	(d) is subject to FCP-EE, and has a valid certification number as follows: _____ (and has not been declared an Ineligible Contractor by HRSD).	
Number of FTEs [Bidders are requested to indicate, the total number of full-time-equivalent positions that would be created and maintained by the bidder if it were awarded the Contract. This information is for information purposes only and will not be evaluated.]		
Security Clearance Level of Bidder *[include both the level and the date it was granted]*		
On behalf of the bidder, by signing below, I confirm that I have read the entire bid solicitation including the documents incorporated by reference into the bid solicitation and I certify that: 1. The bidder considers itself and its products able to meet all the mandatory requirements described in the bid solicitation; 2. This bid is valid for the period requested in the bid solicitation; 3. All the information provided in the bid is complete, true and accurate; and 4. If the bidder is awarded a contract, it will accept all the terms and conditions set out in the resulting contract clauses included in the bid solicitation.		
Signature of Authorized Representative of Bidder		

Government of Canada / Gouvernement du Canada

Contract Number / Numéro du contrat
EN537-05IT01SBIPSG1
Security Classification / Classification de sécurité UNCLASSIFIED

SECURITY REQUIREMENTS CHECK LIST (SRCL)
LISTE DE VÉRIFICATION DES EXIGENCES RELATIVES À LA SÉCURITÉ (LVERS)

PART A - CONTRACT INFORMATION / PARTIE A - INFORMATION CONTRACTUELLE

1. Originating Government Department or Organization / Ministère ou organisme gouvernemental d'origine: Public Works and Government Services Canada	2. Branch or Directorate / Direction générale ou Direction: ITSPD/Aquisitions
3. a) Subcontract Number / Numéro du contrat de sous-traitance	3. b) Name and Address of Subcontractor / Nom et adresse du sous-traitant

4. Brief Description of Work / Brève description du travail

Supply Arrangement for the provision of Solutions Based Informatics Professional Services to the Government of Canada

5. a) Will the supplier require access to Controlled Goods? Le fournisseur aura-t-il accès à des marchandises contrôlées?	✓ No / Non	☐ Yes / Oui
5. b) Will the supplier require access to unclassified military technical data subject to the provisions of the Technical Data Control Regulations? Le fournisseur aura-t-il accès à des données techniques militaires non classifiées qui sont assujetties aux dispositions du Règlement sur le contrôle des données techniques?	✓ No / Non	☐ Yes / Oui
6. Indicate the type of access required / Indiquer le type d'accès requis		
6. a) Will the supplier and its employees require access to PROTECTED and/or CLASSIFIED information or assets? Le fournisseur ainsi que les employés auront-ils accès à des renseignements ou à des biens PROTÉGÉS et/ou CLASSIFIÉS? (Specify the level of access using the chart in Question 7. c) (Préciser le niveau d'accès en utilisant le tableau qui se trouve à la question 7. c)	☐ No / Non	✓ Yes / Oui
6. b) Will the supplier and its employees (e.g. cleaners, maintenance personnel) require access to restricted access areas? No access to PROTECTED and/or CLASSIFIED information or assets is permitted. Le fournisseur et ses employés (p. ex. nettoyeurs, personnel d'entretien) auront-ils accès à des zones d'accès restreintes? L'accès à des renseignements ou à des biens PROTÉGÉS et/ou CLASSIFIÉS n'est pas autorisé.	✓ No / Non	☐ Yes / Oui
6. c) Is this a commercial courier or delivery requirement with **no** overnight storage? S'agit-il d'un contrat de messagerie ou de livraison commerciale **sans** entreposage de nuit?	✓ No / Non	☐ Yes / Oui

7. a) Indicate the type of information that the supplier will be required to access / Indiquer le type d'information auquel le fournisseur devra avoir accès

Canada ✓	NATO / OTAN ☐	Foreign / Étranger ☐

7. b) Release restrictions / Restrictions relatives à la diffusion

No release restrictions Aucune restriction relative à la diffusion ✓	All NATO countries Tous les pays de l'OTAN ☐	No release restrictions Aucune restriction relative à la diffusion ☐
Not releasable À ne pas diffuser ☐		
Restricted to: / Limité à : ☐	Restricted to: / Limité à : ☐	Restricted to: / Limité à : ☐
Specify country(ies): / Préciser le(s) pays :	Specify country(ies): / Préciser le(s) pays :	Specify country(ies): / Préciser le(s) pays :

7. c) Level of information / Niveau d'information

PROTECTED A PROTÉGÉ A ✓	NATO UNCLASSIFIED NATO NON CLASSIFIÉ ☐	PROTECTED A PROTÉGÉ A ☐
PROTECTED B PROTÉGÉ B ✓	NATO RESTRICTED NATO DIFFUSION RESTREINTE ☐	PROTECTED B PROTÉGÉ B ☐
PROTECTED C PROTÉGÉ C ✓	NATO CONFIDENTIAL NATO CONFIDENTIEL ☐	PROTECTED C PROTÉGÉ C ☐
CONFIDENTIAL CONFIDENTIEL ☐	NATO SECRET NATO SECRET ☐	CONFIDENTIAL CONFIDENTIEL ☐
SECRET SECRET ☐	COSMIC TOP SECRET COSMIC TRÈS SECRET ☐	SECRET SECRET ☐
TOP SECRET TRÈS SECRET ☐		TOP SECRET TRÈS SECRET ☐
TOP SECRET (SIGINT) TRÈS SECRET (SIGINT) ☐		TOP SECRET (SIGINT) TRÈS SECRET (SIGINT) ☐

Government of Canada / Gouvernement du Canada

Contract Number / Numéro du contrat
EN537-05IT01SBIPSG1
Security Classification / Classification de sécurité UNCLASSIFIED

PART A (continued) / PARTIE A (suite)

8. Will the supplier require access to PROTECTED and/or CLASSIFIED COMSEC information or assets?
Le fournisseur aura-t-il accès à des renseignements ou à des biens COMSEC désignés PROTÉGÉS et/ou CLASSIFIÉS?
If Yes, indicate the level of sensitivity:
Dans l'affirmative, indiquer le niveau de sensibilité :
[✓] No Non [ ] Yes Oui

9. Will the supplier require access to extremely sensitive INFOSEC information or assets?
Le fournisseur aura-t-il accès à des renseignements ou à des biens INFOSEC de nature extrêmement délicate?
[✓] No Non [ ] Yes Oui

Short Title(s) of material / Titre(s) abrégé(s) du matériel :
Document Number / Numéro du document :

PART B - PERSONNEL (SUPPLIER) / PARTIE B - PERSONNEL (FOURNISSEUR)

10. a) Personnel security screening level required / Niveau de contrôle de la sécurité du personnel requis

[✓] RELIABILITY STATUS COTE DE FIABILITÉ	[ ] CONFIDENTIAL CONFIDENTIEL	[ ] SECRET SECRET	[ ] TOP SECRET TRÈS SECRET
[ ] TOP SECRET– SIGINT TRÈS SECRET – SIGINT	[ ] NATO CONFIDENTIAL NATO CONFIDENTIEL	[ ] NATO SECRET NATO SECRET	[ ] COSMIC TOP SECRET COSMIC TRÈS SECRET
[ ] SITE ACCESS ACCÈS AUX EMPLACEMENTS			

Special comments:
Commentaires spéciaux : _____

NOTE: If multiple levels of screening are identified, a Security Classification Guide must be provided.
REMARQUE : Si plusieurs niveaux de contrôle de sécurité sont requis, un guide de classification de la sécurité doit être fourni.

10. b) May unscreened personnel be used for portions of the work?
Du personnel sans autorisation sécuritaire peut-il se voir confier des parties du travail?
[✓] No Non [ ] Yes Oui

If Yes, will unscreened personnel be escorted?
Dans l'affirmative, le personnel en question sera-t-il escorté?
[✓] No Non [ ] Yes Oui

PART C - SAFEGUARDS (SUPPLIER) / PARTIE C - MESURES DE PROTECTION (FOURNISSEUR)

INFORMATION / ASSETS / RENSEIGNEMENTS / BIENS

11. a) Will the supplier be required to receive and store PROTECTED and/or CLASSIFIED information or assets on its site or premises?
Le fournisseur sera-t-il tenu de recevoir et d'entreposer sur place des renseignements ou des biens PROTÉGÉS et/ou CLASSIFIÉS?
[✓] No Non [ ] Yes Oui

11. b) Will the supplier be required to safeguard COMSEC information or assets?
Le fournisseur sera-t-il tenu de protéger des renseignements ou des biens COMSEC?
[✓] No Non [ ] Yes Oui

PRODUCTION

11. c) Will the production (manufacture, and/or repair and/or modification) of PROTECTED and/or CLASSIFIED material or equipment occur at the supplier's site or premises?
Les installations du fournisseur serviront-elles à la production (fabrication et/ou réparation et/ou modification) de matériel PROTÉGÉ et/ou CLASSIFIÉ?
[✓] No Non [ ] Yes Oui

INFORMATION TECHNOLOGY (IT) MEDIA / SUPPORT RELATIF À LA TECHNOLOGIE DE L'INFORMATION (TI)

11. d) Will the supplier be required to use its IT systems to electronically process, produce or store PROTECTED and/or CLASSIFIED information or data?
Le fournisseur sera-t-il tenu d'utiliser ses propres systèmes informatiques pour traiter, produire ou stocker électroniquement des renseignements ou des données PROTÉGÉS et/ou CLASSIFIÉS?
[✓] No Non [ ] Yes Oui

11. e) Will there be an electronic link between the supplier's IT systems and the government department or agency?
Disposera-t-on d'un lien électronique entre le système informatique du fournisseur et celui du ministère ou de l'agence gouvernementale?
[✓] No Non [ ] Yes Oui

Government of Canada / Gouvernement du Canada

Contract Number / Numéro du contrat
EN537-05IT01SBIPSG1
Security Classification / Classification de sécurité UNCLASSIFIED

PART C - *(continued)* / PARTIE C - *(suite)*

For users completing the form **manually** use the summary chart below to indicate the category(ies) and level(s) of safeguarding required at the supplier's site(s) or premises.
Les utilisateurs qui remplissent le formulaire **manuellement** doivent utiliser le tableau récapitulatif ci-dessous pour indiquer, pour chaque catégorie, les niveaux de sauvegarde requis aux installations du fournisseur.

For users completing the form **online** (via the Internet), the summary chart is automatically populated by your responses to previous questions.
Dans le cas des utilisateurs qui remplissent le formulaire **en ligne** (par Internet), les réponses aux questions précédentes sont automatiquement saisies dans le tableau récapitulatif.

SUMMARY CHART / TABLEAU RÉCAPITULATIF

Category Catégorie	PROTECTED PROTÉGÉ			CLASSIFIED CLASSIFIÉ			NATO				COMSEC					
	A	B	C	CONFIDENTIAL CONFIDENTIEL	SECRET	TOP SECRET TRÈS SECRET	NATO RESTRICTED NATO DIFFUSION RESTREINTE	NATO CONFIDENTIAL NATO CONFIDENTIEL	NATO SECRET	COSMIC TOP SECRET COSMIC TRÈS SECRET	PROTECTED PROTÉGÉ A	PROTECTED PROTÉGÉ B	PROTECTED PROTÉGÉ C	CONFIDENTIAL CONFIDENTIEL	SECRET	TOP SECRET TRÈS SECRET
Information / Assets Renseignements / Biens																
Production																
IT Media / Support TI																
IT Link / Lien électronique																

12. a) Is the description of the work contained within this SRCL PROTECTED and/or CLASSIFIED?
La description du travail visé par la présente LVERS est-elle de nature PROTÉGÉE et/ou CLASSIFIÉE?

☑ No / Non ☐ Yes / Oui

If Yes, classify this form by annotating the top and bottom in the area entitled "Security Classification".
Dans l'affirmative, classifier le présent formulaire en indiquant le niveau de sécurité dans la case intitulée « Classification de sécurité » au haut et au bas du formulaire.

12. b) Will the documentation attached to this SRCL be PROTECTED and/or CLASSIFIED?
La documentation associée à la présente LVERS sera-t-elle PROTÉGÉE et/ou CLASSIFIÉE?

☑ No / Non ☐ Yes / Oui

If Yes, classify this form by annotating the top and bottom in the area entitled "Security Classification" and indicate with attachments (e.g. SECRET with Attachments).
Dans l'affirmative, classifier le présent formulaire en indiquant le niveau de sécurité dans la case intitulée « Classification de sécurité » au haut et au bas du formulaire et indiquer qu'il y a des pièces jointes (p. ex. SECRET avec des pièces jointes).

Government of Canada / Gouvernement du Canada

Contract Number / Numéro du contrat
EN537-05IT01SBIPSG1
Security Classification / Classification de sécurité UNCLASSIFIED

PART D - AUTHORIZATION / PARTIE D - AUTORISATION

13. Organization Project Authority / Chargé de projet de l'organisme

Name (print) - Nom (en lettres moulées)	Title - Titre	Signature	
Toman, Rebecca	Supply Specialist	Rebecca Toman	
Telephone No. - N° de téléphone 819-956-2375	Facsimile No. - N° de télécopieur 819-956-7827	E-mail address - Adresse courriel rebecca.toman@tpsgc-pwgsc.gc.ca	Date Mar 18, 2011

14. Organization Security Authority / Responsable de la sécurité de l'organisme

Name (print) - Nom (en lettres moulées)	Title - Titre	Signature	
Robertson, Julie	SO		
Telephone No. - N° de téléphone 819-956-0615	Facsimile No. - N° de télécopieur 819-934-1449	E-mail address - Adresse courriel julie.robertson@tpsgc-pwgsc.gc.ca	Date 2011-03-21

15. Are there additional instructions (e.g. Security Guide, Security Classification Guide) attached?
Des instructions supplémentaires (p. ex. Guide de sécurité, Guide de classification de la sécurité) sont-elles jointes? ☐ No / Non ☐ Yes / Oui

16. Procurement Officer / Agent d'approvisionnement

Name (print) - Nom (en lettres moulées)	Title - Titre	Signature	
Telephone No. - N° de téléphone	Facsimile No. - N° de télécopieur	E-mail address - Adresse courriel	Date

17. Contracting Security Authority / Autorité contractante en matière de sécurité

Name (print) - Nom (en lettres moulées)	Title - Titre	Signature	
Eric Bouquin	Contract Security Officer		
Telephone No. - N° de téléphone 613 960 9658	Facsimile No. - N° de télécopieur 613 954-4171	E-mail address - Adresse courriel Eric G. Bouquin @ tpsgc-pwgsc.gc.ca	Date March 21, 2011