Copy No. _____

Defence Research and Development Canada
Recherche et développement pour la défense Canada

DEFENCE R&D DÉFENSE

PASTET Development - Use of Sound Speed Profiles

Final Report

Terry J. Deveau
Fred Campaigne
Eric Widdis
JASCO Applied Sciences

Prepared by:
JASCO Applied Sciences
32 Troop Avenue, Unit 301
Dartmouth, NS B3B 1Z1

Contractor's Document Number: 00244
Contract Project Manager: John Moloney, 902-405-3336
PWGSC Contract Number: W7707-098179/001/HAL
CSA: Sean Pecknold, Defence Scientist, Maritime Environmental Awareness, 902-426-3100

Defence R&D Canada – Atlantic

Contract Report
DRDC Atlantic CR 2012-101
July 2012

Canada

This page intentionally left blank.

PASTET Development - Use of Sound Speed Profiles

Final Report

Terry J. Deveau
Fred Campaigne
Eric Widdis
JASCO Applied Sciences

Prepared By:
JASCO Applied Sciences
32 Troop Ave,
Dartmouth, NS Canada B3B 1Z1

Contractor's Document Number: 00244
Contract Project Manager: John Moloney, 902-405-3336
PWGSC Contract Number: W7707-098179/001/HAL
CSA: Sean Pecknold, Defence Scientist, Maritime Environmental Awareness, 902-426-3100

Defence R&D Canada – Atlantic

Contract Report
DRDC Atlantic CR 2012-101
July 2012

Principal Author

Original signed by Terry J. Deveau

Terry J. Deveau

Senior Scientist, Ocean Acoustics

Approved by

Original signed by Dan Hutt

Dan Hutt

Head/Underwater Sensing

Approved for release by

Original signed by Leon Cheng

Leon Cheng

Chair, Document Review Panel

Abstract

DRDC Atlantic has been developing a research-level acoustic prediction system, the System Test Bed (STB), in support of tactical decision aids and improving operator effectiveness. The Environmental Modeling Manager (EMM) is one of the STB components. The Portable Acoustic Sensitivity and Transmission Estimation Tool (PASTET) has been, and is being, developed as an adjunct to the STB and extension of existing acoustic propagation modeling capabilities. The first objective of this call-up is to include the capability for using mean and standard deviation of sound speed profiles in PASTET, along with several bug fixes. The second objective of this call-up is to provide the scientific authority with the tools required to use ocean forecast models as inputs to EMM and PASTET. This report describes the work that was done during the call-up, the issues encountered, the results that were produced, and includes recommendations for additional work.

Résumé

RDDC Atlantique a mis au point un système de prédiction acoustique de recherche, le Banc d'essai de systèmes (*System Test Bed,* STB), dans le but d'appuyer le développement d'aides à la prise de décisions tactiques et l'amélioration de l'efficacité de l'opérateur. L'EMM (Environment Modeling Manager – *gestionnaire de modélisation de l'environnement*) est un des éléments du STB. PASTET (Portable Acoustic Sensitivity and Transmission Estimation Tool – *Outil portable d'évaluation de la transmission acoustique et de sa sensibilité*) est un logiciel mis au point pour servir d'appui au STB et étendre les capacités existantes de modélisation de la propagation acoustique dont le développement se poursuit. Le premier objectif de la présente commande est de permettre l'utilisation de la moyenne et de l'écart type des profils de vitesse du son dans PASTET et de régler plusieurs bogues. Le deuxième objectif est de fournir à l'autorité scientifique les outils nécessaires pour utiliser des modèles de prévision océanique pour fournir des données d'entrée à l'EMM et à PASTET. Le présent rapport décrit les travaux effectués pour cette commande, les problèmes survenus et les résultats produits. Il comprend également des recommandations sur des travaux supplémentaires.

This page intentionally left blank.

Executive summary

PASTET Development - Use of Sound Speed Profiles

Terry J. Deveau; Fred Campaigne; Eric Widdis; DRDC Atlantic CR 2012-101; Defence R&D Canada – Atlantic; July 2012.

Introduction: The Portable Acoustic Sensitivity and Transmission Estimation Tool (PASTET) is a software tool designed to allow transmission loss and sensitivity calculations to be made using environmental data from a variety of sources, and accounting for uncertainty or variability in the environment. This work extends the capability of PASTET to import sound speed profiles (SSP) from DRDC databases and external ocean forecast models.

Results: The PASTET program has been extended to import SSP data from the READB and the PDB; it has also been extended to import temperature and salinity profile data from these databases and to automatically convert it to SSP. Other functional enhancements to the PASTET software were also performed under this call-up, including improvements to the operation of the graphical user interface (GUI), improvements to the interactive map, and improvements to the output displays. This call-up also provided support for sea trial Q343, including importing ocean forecast SSP data into PASTET from external servers over the Internet.

Significance: Access to a sufficient set of SSPs that are accurate representations of the ocean properties and variability of the modelling scenario is essential to achieving meaningful sensitivity analysis results with PASTET. This work has accomplished the software enhancements that provide PASTET with access to that data. This result furthers the goal of the Assessing Sonar Performance in Realistic Environments (ASPIRE) Applied Research Project to develop the capability to determine what the important environmental parameters are for better sonar performance predictions.

Future plans: Providing a robust and portable tool for rapid assessment of sonar performance sensitivity to variations and uncertainty in ocean environmental parameters is the ultimate goal. Additional improvements and enhancements to PASTET's facilities for accessing external data sets, processing the data, performing acoustic modeling, and preparing graphical output products are seen as near-term objectives for continued PASTET work.

Sommaire

PASTET Development - Use of Sound Speed Profiles

Terry J. Deveau; Fred Campaigne; Eric Widdis; DRDC Atlantic CR 2012-101; R & D pour la défense Canada – Atlantique; juillet 2012.

Introduction : PASTET (Portable Acoustic Sensitivity and Transmission Estimation Tool – *Outil portable d'évaluation de la transmission acoustique et de sa sensibilité*) est un outil logiciel servant à réaliser des calculs de perte de transmission et de sensibilité à partir de données environnementales provenant de diverses sources en tenant compte de l'incertitude et de la variabilité du milieu. Les travaux présentés ici augmentent la capacité de PASTET d'importer des profils de vitesse du son des bases de données de RDDC et de modèles externes de prévision océanique.

Résultats : Le programme PASTET a été étendu pour importer des données de profils de vitesse du son à partir de READB et de PDB; il a également été étendu pour importer des données de profils de salinité et de température à partir de ces bases de données et les convertir automatiquement en profils de vitesse du son. D'autres améliorations fonctionnelles du logiciel ont également été réalisées dans le cadre de cette commande, notamment des améliorations de la fonctionnalité de l'interface graphique, de la carte interactive et des affichages de sortie. Cette commande a également servi au soutien de l'essai en mer Q343, notamment pour l'importation de données de profils de vitesse du son issues de prévisions océaniques dans PASTET à partir de serveurs externes sur Internet.

Portée : Pour obtenir des résultats d'analyses de sensibilité valables au moyen de PASTET, il est essentiel d'avoir accès à un ensemble de profils de vitesse du son qui représentent fidélement les propriétés de l'océan et le scénario de modélisation. Les travaux présentés ici ont consisté à réaliser les améliorations logicielles nécessaires pour permettre à PASTET d'accéder à de telles données. Ces résultats contribuent à la réalisation du projet de recherches appliquées ASPIRE (Assessing Sonar Performance in Realistic Environments – *Évaluation du rendement des sonars dans des environnements réalistes*), qui vise à mettre au point une capacité de détermination des paramètres environnementaux importants pour l'amélioration des prédictions sonar.

Recherches futures : L'objectif ultime est de fournir un outil robuste et portable d'évaluation de la sensibilité de la performance des sonars aux variations et à l'incertitude des paramètres environnementaux de l'océan. D'autres améliorations des mécanismes dont dispose PASTET pour accéder à des ensembles de données externes, traiter les données, réaliser la modélisation acoustique et préparer des produits de sortie graphique sont des objectifs à court terme des travaux.

Table of contents

List of figures

Acknowledgements

Fred Campaigne conducted a substantial portion of the work on this project, particularly with regards to the database access functions, and contributed considerable text to this document. Eric Widdis provided quality assurance and document review. Terry J. Deveau, Senior Scientist, Ocean Acoustics, is the primary author of this report and is responsible for the scientific integrity of the contractor work on this call-up.

Sean Pecknold is the overall scientific authority for this project; he is assisted by Meng-Han Chi and Aaron Webb. They provided guidance and direction to the contractor throughout this call-up.

This page intentionally left blank.

1 Introduction

Defence Research and Development Canada – Atlantic (DRDC Atlantic) has been developing a research-level acoustic prediction system the System Test Bed (STB), in support of tactical decision aids and improved operator effectiveness. The Portable Acoustic Sensitivity and Transmission Estimation Tool (PASTET) is being developed as an adjunct to, and extension of, existing underwater acoustic propagation modeling capabilities. The first objective of this call-up 010 under the STB maintenance standing offer contract[1] is to include the capability for using mean and standard deviation of sound speed profiles (SSPs) in PASTET, along with several bug fixes. The second objective of this call-up is to provide the Scientific Authority with the tools required to use ocean forecast model SSPs as input to the Environmental Modeling Manager (EMM) and PASTET. This report describes the work that was done during this call-up, the issues that were encountered, and the results that were produced; it includes recommendations for additional work.

Previous call-ups have progressed PASTET software development, from an MS-Windows application that was embedded within a much larger proprietary third-party application, to a standalone Linux-based system (which can still run under MS-Windows, if desired, using Cygwin). The data access capability was also expanded to use the new DRDC PDB. Additional data access functionality was previously demonstrated by connecting to and retrieving STB-based EMM data (initially, only for one data element). Previous work also demonstrated the ability of a PostgreSQL database to access external NetCDF-based bathymetry data. The recently released ETOPO1 public domain bathymetry data set was also previously integrated with PASTET. From a data and software porting perspective, the most important data integration not completed in the previous work to date is the SSP data element.

The present call-up has accomplished the PASTET integration of SSP data from XBT (Expendable Bathythermograph) and TSD (temperature-salinity-depth) sources using both the READB and the PDB databases. It has also implemented some improvements to the GUI functionality and internal processing capabilities of PASTET, as well as several bug fixes.

In addition, this call-up provided sea trial support for DRDC cruise Q343. This was accomplished by refurbishing software components previously prepared by Fred Campaigne for DRDC cruise Q325, as well as through new engineering. In the present call-up, operational sea trial support software was improvised to import SSP data from the publicly available Mercator Ocean model/forecast TSD predictions.[3] In addition to the software development, the sea trial support provided under this call-up includes set-to-work logistics, remote on-shore operations and data processing during the sea trial, and operational training for DRDC computer scientists who participated in the sea trial.

1.1 Tasking and Scope

Under Call-Up 010 of the STB maintenance standing offer contract, JASCO (the contractor) was assigned the following tasks, to be performed in collaboration with the Scientific Authority.

Task 1 – PASTET work

1. Sound speed profiles will be included in PASTET.
   a. The interface to the READB and Production DB (PDB) will be fixed to enable PASTET to acquire sound speed profile (SSP) data from XBT tables, TSD tables, and ocean forecast model tables.
      i. TSD data tables return depths, temperature, and salinity, from which sound speed can be derived using the formulation found in lines 1454-1652 of `SQLreference.c` (as originally provided).
      ii. Non-physical SSP values will be removed
      iii. SSP will be upward continued to the surface, and downward continued to allow for the maximum bathymetry encountered, using constant temperature and salinity but increasing depth in the formulation described in item i above.
   b. SSPs will be interpolated onto a constant grid (1-m) and the mean determined (point-by-point).
   c. The point-by-point standard deviation of the SSP will be determined, decimated, and used to generate perturbed fields for the PASTET calculations.
   d. If time permits, the empirical orthogonal functions of the set of sound speed profiles and their associated variances will be determined, allowing the construction of multiple perturbed profiles, using the method from the IDL code provided in the previous call-up.
2. PASTET Improvements:
   a. The capability for the user to freely edit input frequency, range, and bearing, and perform full field calculations at receiver depths based on the maximum selected bathymetry rather than at fixed depths will be added.
   e. The work proposed will also result in the implementation of the remaining graphical output products from the original PASTET application; specifically, line plots of sensitivity versus range, full-field plots of TL versus range and depth, and full-field plots of sensitivity versus range and depth.

Task 2 – Support Sea Trial Q343

Technical training and potential interface upgrades or changes will be provided to allow the current MEA group computer scientists to use the capability developed for Q325. Specifically, training and any interface changes required to remotely access ocean forecast models (including one or more of DalCoast, CNOOFS, and NCOM), and to use these in the Environmental Modeling Manager (EMM) and PASTET systems, will be provided and/or implemented. None of the task will be undertaken onboard ship.

The scope of the work is bounded by the $40K fixed budget for this call-up.[1]

2 Implementation

Following the issue of Call-Up 010 under the STB Maintenance standing offer,[1] a kick-off meeting was held at DRDC Atlantic on 21 December 2011. Present at this meeting were DRDC scientific staff Sean Pecknold (Scientific Authority), Meng-Han Chi, Aaron Webb, and JASCO (contractor) staff Trent Johnson and Terry J. Deveau. The logistics of the work were discussed, including the issues pertaining to support for the sea trial, which was planned for departure on 8 January 2012.

2.1 Task 1 — PASTET Work

Prior to the contractor beginning work on this call-up, a DRDC computer scientist discovered an apparent algorithm discrepancy in the source code. Further investigation revealed that a previous PASTET developer had encountered a *stack violations* issue (caused by poor coding techniques) and had attempted to find a work around, resulting in the discrepancy. The present call-up included investigation of the algorithm discrepancy and its correction. The intended original algorithm logic was restored and the coding was improved (including the elimination of a few *goto* statements). The *stack violation* problems were resolved by recoding a number of large arrays to use dynamic memory allocation (through the *malloc* and *free* system calls) and passing pointers as function arguments, rather than allowing the C++ compiler to automatically allocate large chunks of *stack* storage for arrays passed as function arguments.

The statement of requirements for the present call-up identifies three SSP data sources to be made accessible to PASTET:

1. XBT tables,
2. TSD tables, and
3. Ocean forecast model tables.

Also, the direction from the DRDC Scientific Authority was that both the READB and PDB databases were to be made available for access by PASTET. The question of how to handle and process ocean model forecast SSP data was examined as part of the cruise Q343 sea trial support (see the next section). Although access to this data was provisionally obtained for the sea trial, a permanent solution for regular PASTET access to the forecast SSP data was not readily forthcoming—for logistical, network, and security reasons that the contractor cannot control. The approach taken was to get the READB functionality operational for both XBT and TSD first; once these were working, their SQL code could be reworked to apply to the PDB—the PDB schema [2] is considerably different from the READB.

Therefore, once the integration of the XBT and TSD sound speed data was completed for the READB, work began on converting the SQL code to a version that would be suitable for the PDB. An example of the coding of an SQL query for READB TSD data is shown in Annex A, and the coding for the equivalent PDB query is shown in Annex B. Similarly Annex C provides an example of a READB XBT query and Annex D shows the current PDB equivalent.

The PASTET GUI was also modified so that the four SSP data source selection choices are offered in the drop-down box on the SSP panel.

A deprecated SSP query was discovered in the READB that used CNOOFS 0.25 degree model/forecast data. This query used the PDB table structure from within the READB. It was developed on cruise Q325 by DRDC scientist Anthony Isenor. The details of this SQL query would be viable as an example for accessing model sound speed data from within the PDB.

An error was discovered in the great-circle utility that was treating the bearing angle incorrectly, resulting in survey data being returned for the wrong LOBs. The bearing angle was being passed into the function already formatted as CW degrees from true north; however the function was then converting it to CCW degrees from due east, then passing this converted bearing into a numerical code block that was actually expecting bearing as CW degrees from true north. This was analysed and corrected by removing the extra, inappropriate conversion.

An interim PASTET release was installed at DRDC Atlantic after the work described above was completed. This provides an opportunity for DRDC scientific staff to test the work that was accomplished to that point and to give preliminary feedback to the contractor. This DRDC testing revealed a defect in bathymetry data access for longitudes east of Greenwich. The problem was tracked down to an inconsistency in the order in which the arguments were passed between two routines—describing the extents of the grid for the data base query. This problem was analysed and corrected by rectifying the argument sequence to be compatible in both routines.

Subsequent to this, the remaining software change requirements were implemented (where possible), and in the process of doing so (as well as in testing these changes), some additional pre-existing defects in the software were discovered. These were analysed and fixed, where possible, and where sufficient resources were available in the present call-up to do so; otherwise the problems were noted, to be addressed in future work.

The *frequency*, *range*, and *bearing* input fields were changed in the GUI from menu selection objects to freely-edited text input fields. Other miscellaneous source code cleanups were addressed in the process.

It was discovered that a *malloc* error could occur in the bathymetry query code when `Run` is attempted without first doing `Survey`. This was analysed and corrected. A number of other potential memory management issues (related to *malloc* and *free* calls, and pointer initialization) were identified and corrected in the process. The PASTET program now allows `Run` to be used without doing `Survey` first, and in this manner the manually-entered (or default) parameter values from the GUI screen are used for the run.

The requirement was implemented for the PASTET receiver depths to range from the ocean surface to the deepest point in the set of bathymetry profiles returned by the survey, instead of down to an arbitrary fixed-depth limit of 5000 m. Some issues were also found and corrected for *malloc* and *free* system calls, and memory pointer initialization, in SSP processing.

A bug was corrected where changing the modelling location via the map was not resetting the survey results in the same way as changing it via keyboard entry of the location. The location-dependent GUI fields are now reset to screen-based values (as opposed to survey-based

values) when the latitude/longitude of the modelling location is reselected using the map pointer (this was previously the case when the latitude/longitude was changed via the keyboard).

An error was corrected that was producing "NaN" in the default average sound speed when bathymetry data was found in a survey, but SSP data was not found. This was analysed and corrected.

The pre-existing code has a `max_depth` variable to filter SSP queries to return only SSP depths that are less than or equal to `max_depth`. This filter is supposed to be able to be disabled by passing `max_depth=0`. The logic for this had errors and has now been corrected. The pre-existing code was implemented using `max_depth=1000`. This is not appropriate and the current code now uses `max_depth=0`.

A new variable `dMaxDepth` has been introduced to define the depth to which the SSP data will be extended, or at which it will be interpolated and truncated, whichever is appropriate. New code has been developed to accomplish this, in accordance with the requirement for extending the SSP data to the ocean bottom. Source code was also developed to extend the SSP upwards to depth=0 if necessary.

The requirement for the removal of non-physical SSP values was completed. New constants for SSP quality control are defined as `MAX_SOUND_SPEED=1620`, and `MIN_SOUND_SPEED=1420` (in m/s). A default salinity of 35 PPT was added to the SSP data query logic so that XBT data (not having any salinity) can be converted to SSP data. The use of the existing `NO_VALUE` defined constant was standardized everywhere in the source code (instead of occasional use of "-1" also).

A bug was discovered, and corrected, that continued attempting to adding depth points to last SSP once the maximum number of SSP's was reached.

The calculation of the bathymetry statistics `zMu` and `zSigma` (mean depth and standard deviation, as a function of range, respectively) was moved in the source code from after the survey results to before the SSP survey, so that the `dMaxDepth` parameter for input to the SSP survey can be derived from `max(zMu+zSigma)`.

A bug was corrected where the `dvSSPDepthNewMuPlus` vector was not being set from GUI entered values, when it should have been.

Analysed and corrected memory management issues in the bottom survey routines (`REA.c` and `SQLReference.c`), and made changes to allow the latitude/longitude positions of the bottom survey results to be returned to the top-level PASTET code.

Corrected a number of instances in the PASTET code where floating point equality/inequality tests were used (changed to greater/less than tests).

Changed the `despike_ssp`, `decim_ssp`, and `compress_ssp` functions in `SQLReference.c` to update all three temperature/salinity/sound-speed vectors instead of just the primary vector (this is necessary for SSP profile extension using constant temperature).

Added options on the PASTET GUI map to display the locations of the SSP, bathymetry, and sediment properties survey results.

Implemented the fixed-grid SSP averaging option; this involves populating a fixed-grid structure, profile interpolation on to the grid, calculating the mean gridded SSP and standard deviation, and decimation onto a reduced grid. The initial grid uses a 1 m spacing for the first 900 points, and 50 m spacing thereafter, to a maximum of 1024 points (i.e., a maximum depth of 6950 m). The decimation is not performed if there are 64 points or less in the fixed-grid average SSP. The decimation factor is set to either 3, 5, 10, or 20, whichever is the minimum required to bring the number of profile points to less than 64.

Corrected bugs in the previous sampling-based SSP averaging option (also called the "new" or "experimental" averaging option). Renamed the three SSP averaging options on the GUI to "Vertex", "Sampling", and "Grid", respectively referring to the algorithms on which they are based. Moved the SSP averaging option radio buttons outside the SSP survey block, to clarify that their influence is applied after, and independent of, the survey operation. Enhanced the SSP plot to correctly display the SSP results of the three averaging options.

Changed the GUI default SSP standard deviation to 5 m/s. Altered the SSP averaging logic to utilize this GUI SSP standard deviation value (whether default or manually entered) wherever the SSP survey results are inadequate to calculate a standard deviation (i.e., where fewer than two sound speed values are found for any given average SSP depth).

Implemented the line plot of sensitivity versus range and modified the main GUI page for the new button to launch this new plot.

In summary, all the tasks that are outlined in the statement of requirements for this call-up were completed, except for the following (due to insufficient project resources and the budgetary price ceiling):

a. Task 1.1.a (last part), the PASTET implementation of SSP acquisition from ocean model forecast tables in READB and PDB.

b. Task 1.1.d, the implementation of the Empirical Orthogonal Function (EOF) code and the Monte-Carlo perturbation method.

c. Task 1.2.b, (last part) the implementation of the full-field plot of sensitivity versus range and receiver depth, and the full-field plot of transmission loss versus range and receiver depth.

2.2 Known Unresolved Bugs

1. Near the end of the call-up it was discovered that some bad XBT data is being returned by the SSP survey (duplicate depth points with different temperatures); this has not yet been fully investigated.
2. If the SSP plot window is open, and the operator changes the SSP averaging radio button setting, the SSP plot should automatically refresh to reflect the new selection. In the current PASTET release it is necessary to close the SSP plot window, and re-open it, to display the newly selected SSP averaging.

3. In testing the SSP survey operations, there were a few rare combinations of parameters that caused the survey to hang, and never return. One example was near Emerald Basin, with 2-degree square area, and the READB/TSD data source. This issue requires investigation.
4. The "# samples" GUI field is supposed to be set to "0" when the SSP GUI parameters are manually altered. Currently, it is being set to "1", which is misleading and incorrect.
5. If the map window is open with display of survey results locations selected, and the operator performs a new survey operation, the map should automatically refresh to reflect the new results when the survey operation completes. In the current PASTET release it is necessary to close the map window and re-open it, or perform any other mapping operation, such as zoom or pan, to display the new survey results locations.
6. In testing the mapping of the survey results, for the default modelling location and both survey areas set to 2-degree squares, if the initial map with the three survey results locations displayed is zoomed-in by four clicks of the mouse thumbwheel, a *segmentation fault* is reported—even if ample time is allowed for the screen to refresh between the thumbwheel clicks. This requires investigation.
7. In testing the bathymetry survey results, it has been observed that bathymetry profiles are spaced in range at approximately 12 km between points. The underlying bathymetry grid is much denser than this (a few 100 m) so it begs the question as to why the bathymetry grid is being sampled so sparsely (intentional design or obsolete hardcoding). Other than noting the behaviour, no investigation has yet been attempted.
8. In testing the bathymetry survey results, three of the four available bathymetry data sources appear to return depth values that are consistent with each other, however the fourth one (ETOPO5 from READB) appears to return bathymetry results that are drastically out of line with the other three sources. This requires further investigation.
9. In testing the sediment properties survey results, for certain modelling locations and a large survey area (e.g., a 2-degree square), a number of sediment type results can be reported. However, several of these are duplicates, with the same latitude, longitude, and code value. The reason for this duplication has not yet been investigated. It may be trivial or it may portend a more serious logic error in the way that the sediment type survey results are handled.
10. There is a bug in the text field GUI object. If the full contexts of the text field are highlighted (turned to white-on-black instead of black-on-white) and a character then typed on the keyboard, PASTET will abort with a *segmentation fault*. This needs to be investigated and resolved.
11. Bellhop aborts with errors when the supplied SSP isn't as deep as the deepest supplied bathymetry. This aborts the run without producing any results files, and a message is displayed explaining that no run results can be found. This error should no longer occur with the current PASTET release because the SSP is now ensured to be as deep as the deepest bathymetry (it wasn't in earlier releases). However, in the past, it was occasionally noted that rather than returning from the run with no results, Bellhop failed to abort but merely hung, and therefore the run never returned at all, but remained hung as well. While one expects that this will no longer occur due to SSP data that isn't deep enough, it could still occur due to some other Bellhop error. An investigation should be undertaken to ascertain why Bellhop hangs in these cases, rather than abort as it should, and the problem corrected if possible.

12. Bellhop uses a `temp` directory to store its working files. This directory is created automatically the first time Bellhop is run after a new PASTET installation, if it doesn't already exist. However, the first Bellhop runs still fail for some reason. Subsequent runs are fine. This should be investigated and corrected.
13. A number of "print statements" remain embedded in the PASTET code, left over from various testing efforts. These should be removed or commented-out, as they have outlived their current usefulness and merely add clutter to the `xterm` window where PASTET was launched.

2.3 Suggested Enhancements

1. There are two menus by which the source of the bathymetry survey data is selected, one menu for ETOPO1/2, and a separate menu for READB/PDB. The ETOPO1/2 specification should be combined with the READB/PDB menu in the bathymetry survey block (similar to how the SSP survey source menu is handled) and the separate ETOPO1/2 menu eliminated.
2. Since the survey results are accessed sequentially as bathymetry first, then SSP, and finally sediment properties, it may be appropriate to reorganize the GUI main screen to portray the corresponding survey blocks in that same order from top to bottom. This isn't a "bug", as it will not affect the results either way, but it may make the PASTET survey operations easier for the program user to visualize and understand. It is a suggestion offered for consideration.
3. Under the present PASTET operational procedures, a run can be performed without first doing a survey (i.e., using all manually-entered, or default values, from the GUI screen), or after a survey. If any GUI field is altered manually after a survey, all the survey results *that pertain to it* are discarded, and if a run is then performed it will use a mix of survey results and GUI specified parameters. The "# samples" GUI field is used to portray whether GUI specified parameters apply for the current run (samples=0), or whether the survey results apply (samples>0). A problem occurs, however, if the operator wants to retain hand-entered data in one group (such as bathymetry, for example), but resurvey another group (such as SSP for a different month, for example). At present every resurvey overwrites all hand-entered GUI data. There is a simple solution to this problem: add the option "none" as one of the menu choices for each of the three survey data sources. When a survey is requested, would not be performed for data sources selected as "none"; instead, these would preserve their GUI values, and those values would be used in the next run, along with the new survey results.
4. In the present PASTET release, SSP extension of survey results is performed to a depth dependent on the deepest bathymetry mean+sigma survey return. In the case where the SSP survey is being performed, but the bathymetry is not being surveyed (GUI field parameters being retained), as described above, the SSP extension must be performed to a depth dependent on the GUI field parameters mean+sigma.
5. It would be useful to provide the map with the capability to display a pop-up screen with the details of the sediment properties, SSP, or bathymetry when the operator left-clicks the pointer on any such corresponding symbol displayed on the map.
6. Implement the action of the keyboard "tab" and "shift-tab" keys for focus navigation on the GUI screen. Currently these keys have no effect.

7. As a direct result of porting PASTET to C++, numerous compiler warnings are generated. Although no warnings, so far, have led to erroneous software, they should be rectified. Deprecation based warnings are the most important to rectify. Leaving these may result in compile failures with new C++ compiler releases. This would also greatly assist in future maintenance and enhancement efforts. Some of this was accomplished in the recent two call-ups, but it is not yet complete.
8. The PASTET source files should be renamed to use the `cpp` suffix. This will enable easy C++ detection for various development environments.

2.4 PASTET Source Code Installation and Development Notes

PASTET is now a C++ application and easily integrated into the Eclipse Software Development Environment.[4] The following tips provide advice to get the most from the Eclipse environment with PASTET:

a. Ensure that the CDT plug-in is installed in Eclipse;
b. Start a new *C++ makefile* project, pointing to the PASTET root source directory;
c. Add `*.c` files to the `Preferences->C++->FileTypes` options list.

The installation of a complete development system is now very simple. This procedure should allow easy setup with all development libraries. On a clean Ubuntu 10.04 system, the following libraries are required. The Synaptic Package Manager may be used to simply the installation of the following:

1. build-essentials
2. libx11-dev
3. postgres-client-common
4. libpq-dev
5. libxt-dev
6. libxaw7-dev
7. libgeotiff-dev
8. libtiff4-dev
9. pgadmin3
10. gfortran

2.5 Task 2 — Support Sea Trial Q343

Support for the cruise Q343 sea trial (January 2012) on CFAV Quest took immediate priority at the start of the call-up; and Task 2 was actually accomplished before Task 1, described in the previous section. The sea trial support requirement for this call-up is essentially to obtain forecast SSP data daily while at sea. Previously on cruise Q325, three SSP forecast data sources were used (DalCoast, CNOOFS, and NCOM); however, the current availability of access to these sources

was uncertain at the time of preparation for the Q343 sea trial. The contractor was directed to prepare to attempt access to NCOM and CFNOOS forecast SSP data; in the meantime the Scientific Authority would investigate their actual availability.

During the time elapsed since cruise Q325, the DREnet on CFAV Quest and ashore at DRDC has implemented tight network security restrictions and the software previously engineered for cruise Q325 will not work anymore. Specifically, the ability to download data into DREnet using FTP is now denied. As a work-around to this issue, operational access to an external website was obtained and a `cron` job was implemented to automatically download test data via FTP. Once the binary NetCDF data was available, a PHP CGI script to MIME-encode the binary into ASCII text was developed and tested. The purpose of the MIME encoding into ASCII text is to make the data more acceptable to the automated DREnet security system scanners, which quickly filter-out binary data sets as potentially unsafe. Scripts for both MIME-encoding and decoding were developed. However, following this work, the Scientific Authority discovered that the CNOOFS and NCOM data sets would not be available anyway. The scripts are included with the software delivery for this call-up, as they may be useful in future work of this nature.

As the departure date for cruise Q343 approached, the Mercator Ocean forecast SSP data[3] was identified as the new source for the SSP ocean model data. A simple system to acquire this data daily in day-1, day-2 and day-5 predictions was implemented through the Mercator website. During the sea trial, the contractor on-shore would operationally upload these three data sets to the external website and the sea trial personnel onboard ship would download the compressed data via their HTTP satellite connection. This system is simple and was completely successful.

The complete software support environment for C++, PostgreSQL and netCDF development, file manipulation and access was loaded and set-to-work on the computer system (`Antec3`) that had been designated for the sea trial. This also includes copies of all reference documentation for netCDF and the support utilities. A third party file viewer was also installed.

In summary, the sea trial support for cruise Q343 that was provided under the present call-up includes the following:

1. Developing an external network system to acquire NCOM and CNOOFS (to be used in the event that these data sources became available for the sea trial);
2. Developing software to read Mercator Ocean NetCDF data;
3. Developing a network system to make the Mercator Ocean data available to the high security DREnet through satellite HTTP;
4. Resurrecting code developed on Q325 to enable the Mercator Ocean data to be written to both the READB and EMM; and
5. Daily uploading Mercator Sound Velocity data for the Q343 Operations Area to an independent third party website.

2.6 SSP Acquisition Software Development Notes

A copy of the complete source code that was used to populate the SSP data into both the EMM and READB during Q325 was also included for the Q343 sea trial. (This provides a template to

facilitate the integration of the Mercator SSP data). The `Antec3` machine has all of the PostgreSQL libraries; a C++ program need only use the `#include "libpq-fe.h"` source code header file and the `-L"/lib/postgresql"` compile command linker switch to gain access to the shared library PostgreSQL functionality. All PostgreSQL and NetCDF programmer documentation files in PDF format are also installed on the `Antec3` computer, in the `Q325 archive` folder.

The Mercator Ocean NetCDF data is public, stable and well documented. The format is easily viewed using the NetCDF operator `ncdump` routine. Using this format, a C++ *object* was written to facilitate access to the Mercator data. Similar NetCDF software was generated for NCOM and CNOOFS data for cruise Q325. This *object* was used in the EMM to populate both the EMM data items and the READB.

To use the Mercator Ocean object, refer to the `test_main.cpp` source code file in the `src` directory. There are comments in the source code itself that simplify its use. The `make` file should be modified by commenting out the executable output directive and un-commenting the library output directive. Using a library will make it easier to integrate with the STB.

The complete set of C++ files used on cruise Q325 to populate both the STB and READB was uploaded to `drdc.gwenzor.com/q343/q25/` subdirectories. This source code was downloaded to Quest and used as a template to build the Mercator Ocean client.

The STB has a system for its *clients* to read data files. In `ncdm.cpp` the `read_file()` method parses the `ncdm.cfg` file. This file should available in the DRDC repositories for Q325 check-out and a copy is on `drdc.gwenzor.com`. The vast majority of this work will not be required. The directory where the NetCDF files reside, and the file naming conventions, are probably the only entries that require modifications.

3 Results

3.1 Enhanced PASTET GUI

The new control page for the PASTET GUI is shown in Figure 1. This view depicts the state of the GUI after the `Survey` and `Run` operations have completed. The four `Plot` buttons at the bottom centre of the page, along with their modifier menus on their left, are only present after the `Run` operation is complete.

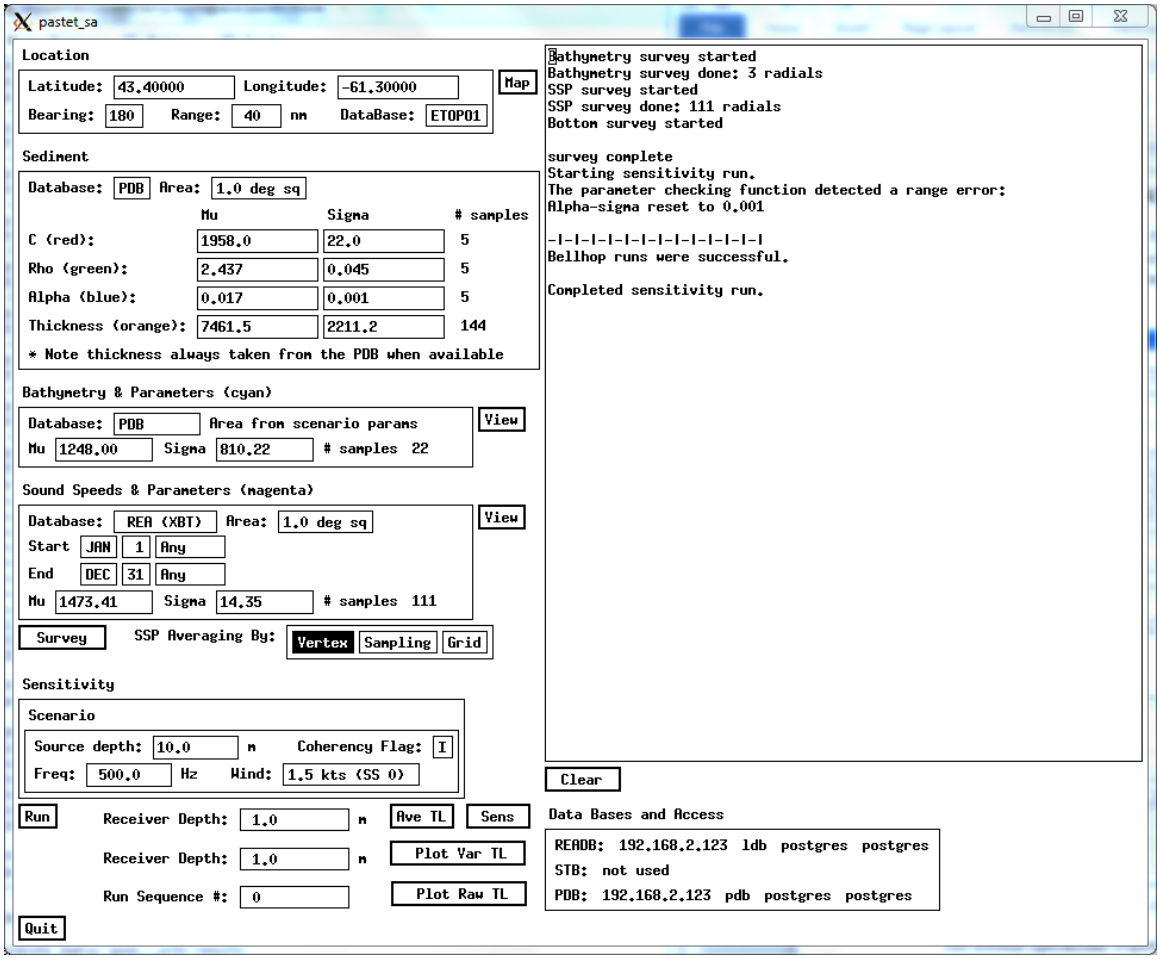

Figure 1: New control page for the PASTET GUI.

The normal operational sequence is for the operator to first specify a latitude-longitude point of origin and LOB in the `Location` box (upper left). The latitude-longitude can either be hand-entered in the corresponding text fields, or graphically selected using the `Map` button (described below). The `Bearing` and `Range` are free-format text fields that can be entered and edited using the pointer and the keyboard. The `Database` control is a pop-up menu from which one value can be selected in a list of choices.

Next, the operator may wish to specify the Sediment box Database and Area controls in preparation for the Survey. These determine which database the Survey will query for the sediment data, and what size geographic square centred on the origin will be considered in the search. The default values will often be correct, and if a selection is made, it remains in effect until the program is shut down (or until another selection is made). The same comments apply to the Database menu controls in the Bathymetry and Sound Speed boxes.

The Bathymetry box doesn't have an Area control. That's because the bathymetry search area is controlled by the LOB bearing and range. Three (or more) bathymetry profiles are taken, one along the specified LOB, and additional ones along radials on either side of it (the radial spacing is database dependent, and more radials are used for longer ranges).

The Sound Speed box does have an Area control, and it also has a start/end date control. These controls allow the operator considerable flexibility in filtering the SSPs that will be included in the averaging calculations. Once the filter controls described above are set to the operator's satisfaction, the next step is to press the Survey button. PASTET then submits a set of queries to the selected databases, calculates the means (Mu) and standard deviations (Sigma) for each of the parameters, and updates the GUI text fields with the results.

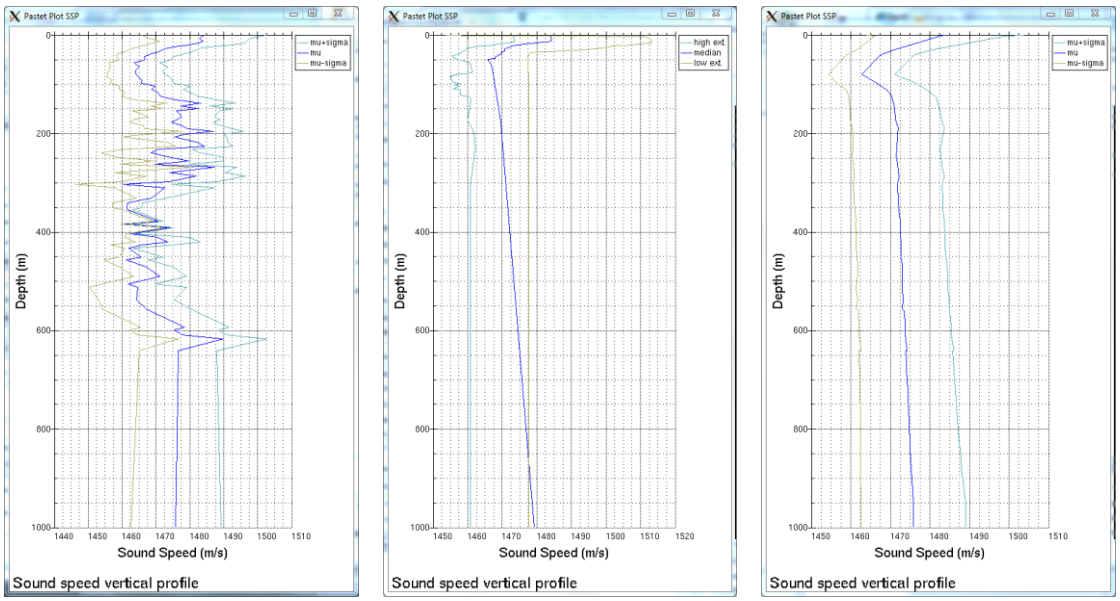

Figure 2: Examples of the average SSP plots that are displayed using the View button, for each of the three SSP averaging options: Vertex, Sample, and Grid, respectively.

In the SSP case, an averaging method selection is also available: (a) Vertex-based SSP averaging, (b) a Sampling approach where the median profile is selected along with the two most extreme examples higher and lower than it in sound speed, and (c) a uniform Grid-based averaging method that is new with this release of the PASTET software. The View button can be clicked to pop-up a window showing the result of the SSP averaging. **Error! Reference source not found.** shows an example of the three different averaging methods on the same survey results. The averaging method selection can be changed before or after the Survey operation; it

doesn't affect the operation of the `Survey`, only how the `Survey` results are subsequently processed.

While the survey is in progress, informational messages are shown in the log box in the upper right half of the GUI window. The `Clear` button can be used to erase the contents of the log box if desired. In addition to the log box message "`survey complete`", the operator is alerted that the survey is finished by the `Survey` button changing colour from black to white.

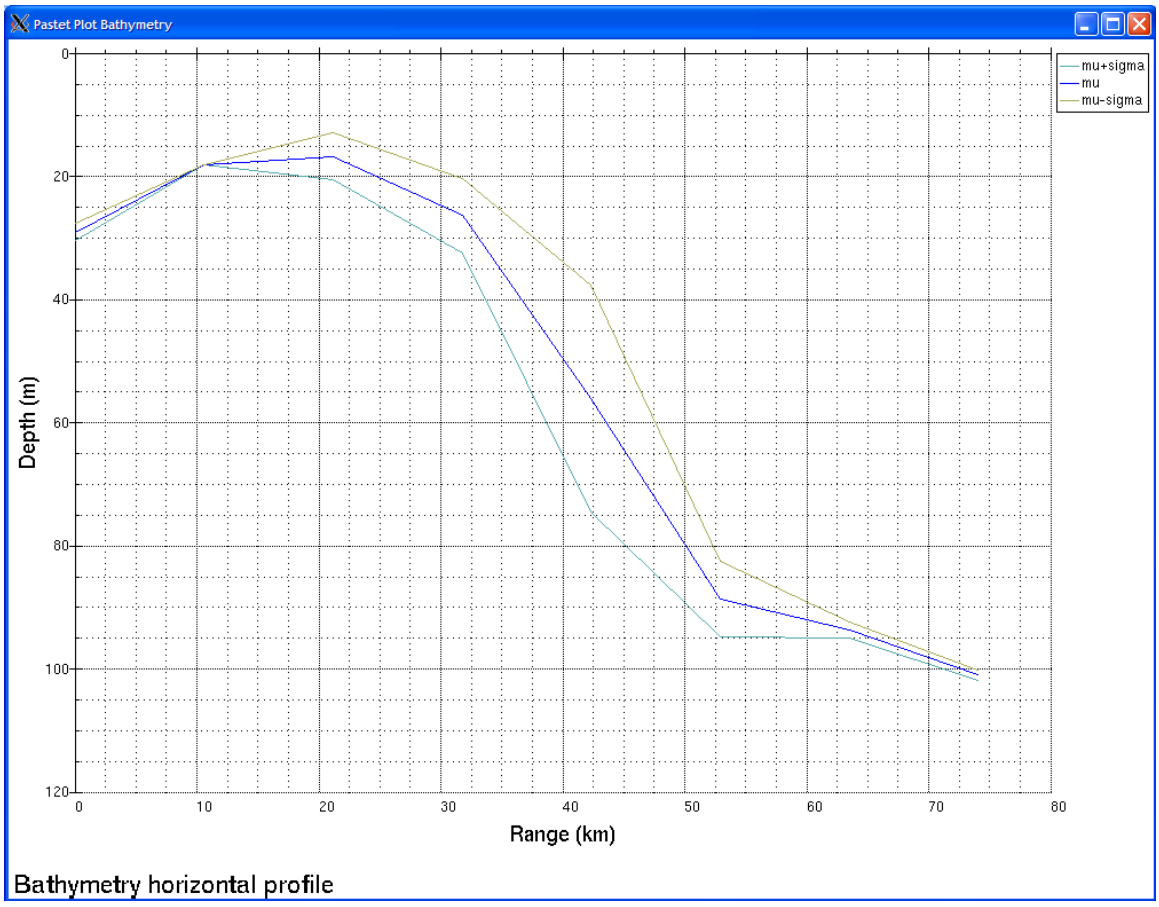

Figure 3: Example of `View` plot of an internal PASTET average bathymetry profile, with plus sigma and minus sigma variations.

Although the PASTET GUI only shows one overall `Mu` and `Sigma` for each of the bathymetry and sound speed profiles, for simplicity of display, it actually has an individual `Mu` and `Sigma` for each point along the profile, internally. These internal `Mu` and `Sigma` profiles can be shown by clicking on the corresponding `View` button. Examples of a PASTET `View` of the internal `Mu` and `Sigma` profiles for sound speed are shown in **Error! Reference source not found.**, and bathymetry in Figure 3.

Following the completion of the `Survey` operation, the operator would typically click on the `Run` button to begin the TL and sensitivity calculations. The operator does have the ability,

however, to manually adjust any of the parameters prior to starting the `Run`. In fact, the operator can use entirely hand-entered data, if desired, and start the `Run` without having done a `Survey` at all. However, there is no way to hand-enter complex bathymetry profiles and sound speed profiles. If the operator edits these GUI fields, the nominal hand-entered profile will be used instead of the survey-derived profile (if any). In the case of bathymetry, the hand entered profile consists of the same depth (`Mu`) at every range point (plus and minus the same `Sigma`), and the hand-entered SSP consists of the same isospeed (`Mu`) at every depth (plus and minus the same `Sigma`). If the operator does edit these fields, and the hand-entered data is thereby selected, the operator is alerted that this has happened by the corresponding "`# samples`" figure being reset to 0. If it was done by mistake the operator can correct it by re-doing the `Survey`.

Prior to selecting `Run`, the operator can also specify the source depth, frequency, wind speed, and coherency option in the `Scenario` box. Each of these settings will affect the TL and sensitivity results, but PASTET is not presently set up to calculate sensitivities for these parameters.

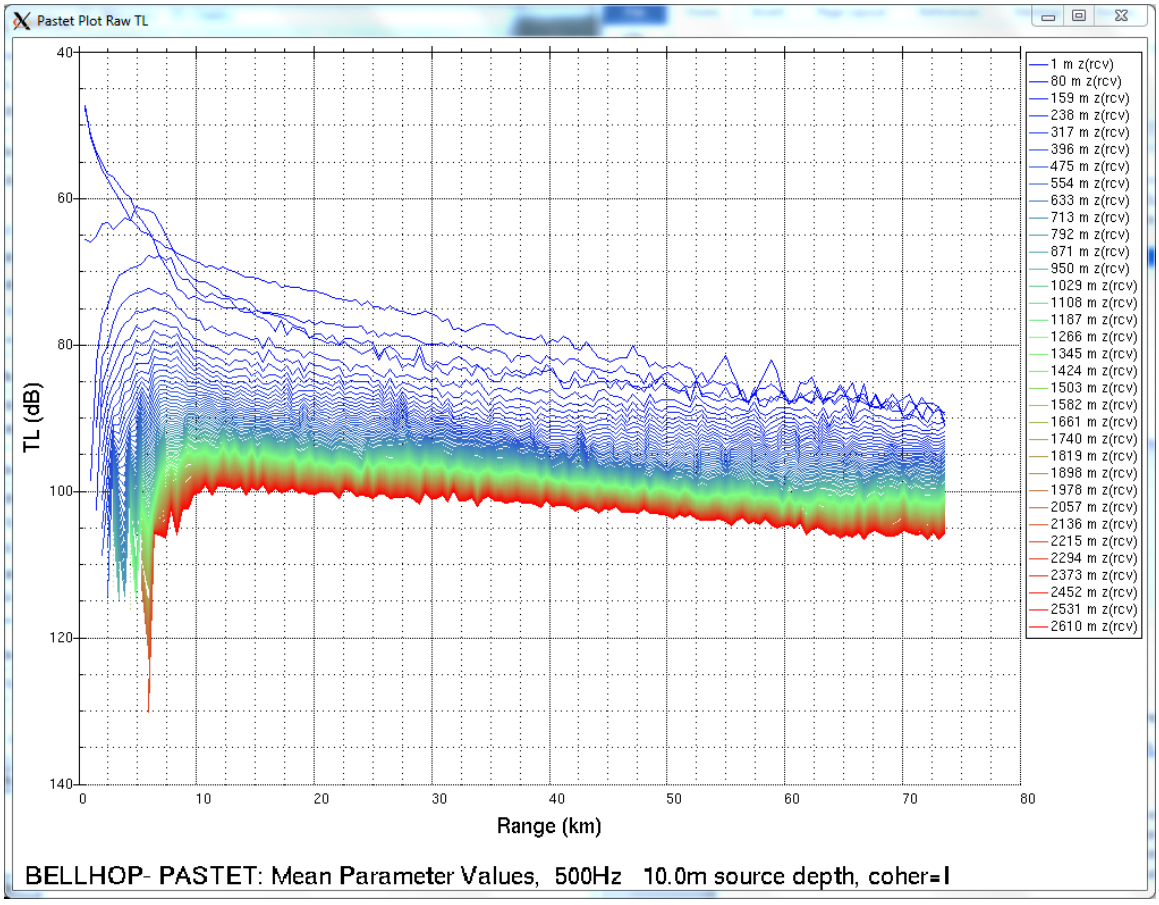

Figure 4: Example of `Raw TL` plot.

Once `Run` is clicked, PASTET begins to sequentially call the Bellhop model to calculate TL versus range and receiver depth for the mean parameter case, and individually plus and minus sigma for each of six parameters (thirteen runs in total). Additional progress messages are written

to the log box during this time. In addition to the log box message "`Completed sensitivity run`", the operator is alerted that the run is finished by the `Run` button changing colour from black to white and the appearance of four `Plot` buttons and related controls to the right of the `Run` button (see Figure 1).

3.2 Improved Output Products

In addition to the bathymetry and SSP profile `View` output products described above (see **Error! Reference source not found.** and Figure 3), there are three types of TL plots available. The `Average TL` and `Variable TL` plots each include a receiver depth selection menu. A file selection menu is provided for the `Raw TL` plot—the "`Run Sequence #`" menu allows the operator to specify which of the 13 TL files from the latest run to be selected for plotting. A concise description of the particular file selected is printed along the bottom edge of the plot itself (see Figure 4), if the selection by sequence number is a bit cryptic. The sequence number selection control has the advantage of ensuring an exact correspondence with the file name on disk.

Figure 5: Example of `Ave TL` plot, showing the mean TL curve and the maximum and minimum envelope curves due to the plus and minus sigma perturbations of all the input variables.

The Raw TL files have 100 receiver depths; these are decimated to 34 curves for the purpose of the plot. This is partly because doing so visually improves the look of the plot, but also because the plotting utility currently in use has a maximum of 34 curves per plot.

An example of the plot displayed by the "Ave TL" button is shown in Figure 5. In addition to portraying an overview of the minimum and maximum TL variations above and below the mean TL, for all input perturbations considered together, it also has been "averaged" (i.e., smoothed) in range and receiver depth.

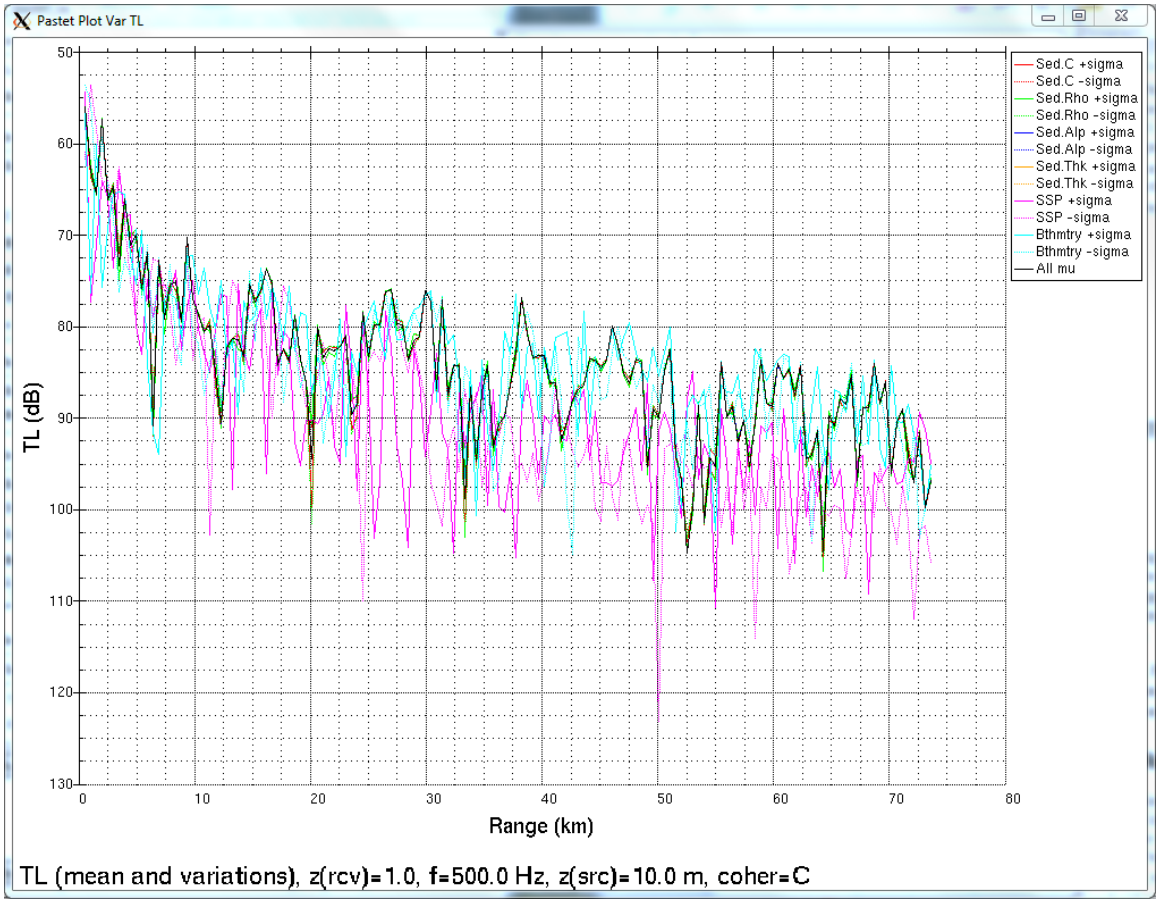

Figure 6: Example of Var TL output product, run using coherent pressure summation, instead of incoherent as in the previous figures.

The third type of TL plot available in PASTET is the "Plot Var TL" output product, an example of which is shown in Figure 6. Here the full-resolution TL is shown (not averaged or decimated in range or receiver depth) and 13 separate curves show the mean and the +/- sigma variations of each parameter. Note that this particular example was run with the coherency option set to "C" (coherent pressure summation), as opposed to the previous figures that showed the results of the same case, except for the selection of coherency option "I" (incoherent pressure summation). When the Var TL plot is shown for the incoherent case, it looks quite similar to Figure 5.

A new graphical output product is now available with the current PASTET release: the sensitivity line plot, in dB versus range for each of the six statistical variables. This plot shares the same receiver depth selection menu as the Average TL plot (and the two buttons are side by side on the main GUI window). An example of a sensitivity plot is shown in Figure 7. Although shown in dB in the plot, the sensitivity is calculated internally in linear units, and is sometimes zero; since a value of zero cannot be portrayed in a dB plot (i.e., it would be negative infinity in dB) gaps will appear in the line plot where the value is calculated as zero—this can be seen, for example, in the sediment density sensitivity (green line) in Figure 7.

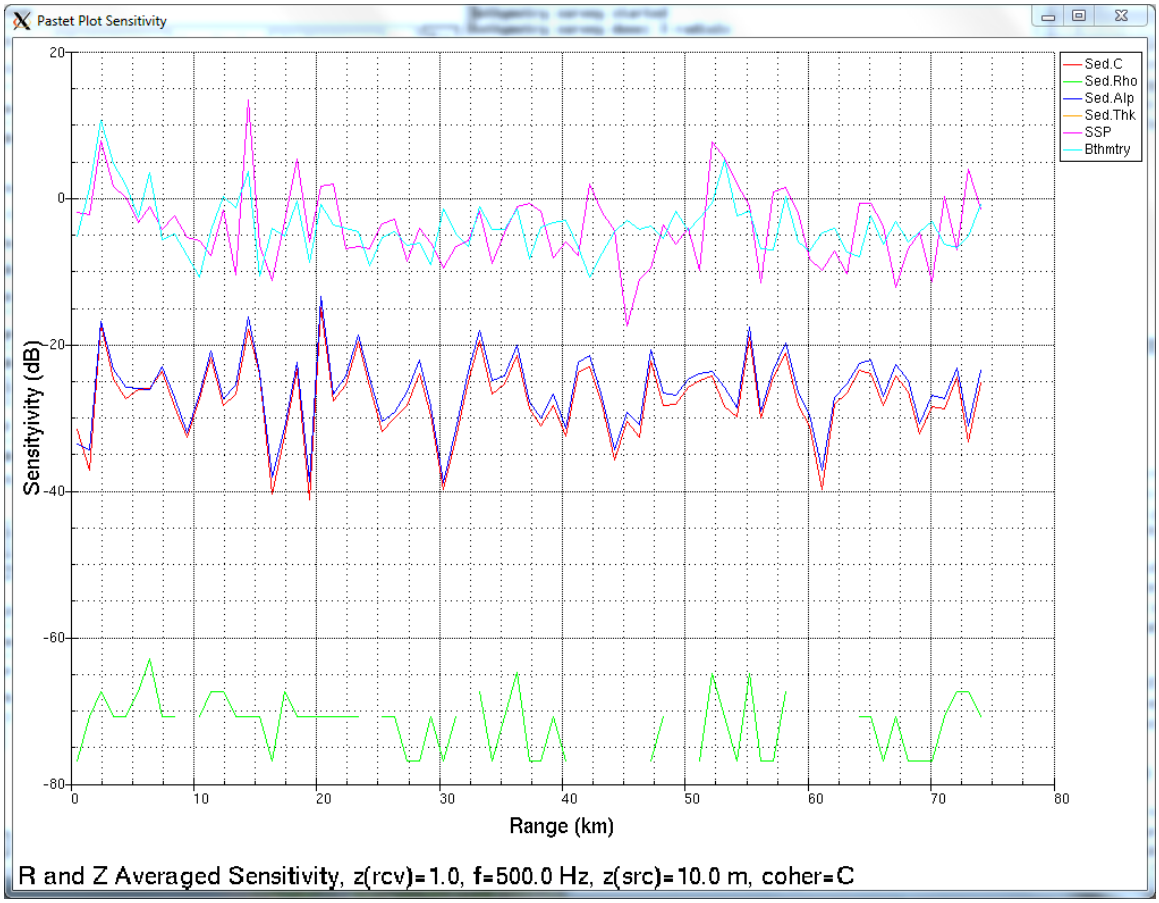

Figure 7: Example of sensitivity line plot, in dB versus range, for each of the six statistical variables.

Each of these plots can be opened by clicking on the corresponding "Plot" button, of course. The button turns black while the plot is open. The plot can be closed by clicking the button again (it is a toggle), or by clicking the "X" in the upper right corner of the window. If another `Run` is started, all the open plot windows will be automatically closed, and even the plotting controls will be removed until the new run is complete.

In addition to the graphical output products, PASTET writes a full copy of all input and output variables and values for every run to a comma-separated-values (CSV) disk file. The previous files are automatically renamed to serve as an archive. The file name is `PASTET_runs.csv`.

3.3 Enhanced Interactive Map

The interactive map was present in previous versions of PASTET. It includes a map projection menu and options for axis labelling and grid lines. An example of the map is displayed in Figure 8. The map is opened when the operator clicks on the `Map` button, which then changes colour to black to indicate that the map is open; the map can be closed either by clicking on the `Map` button again (it is a toggle), or by clicking on the "X" in the top right corner of the map window.

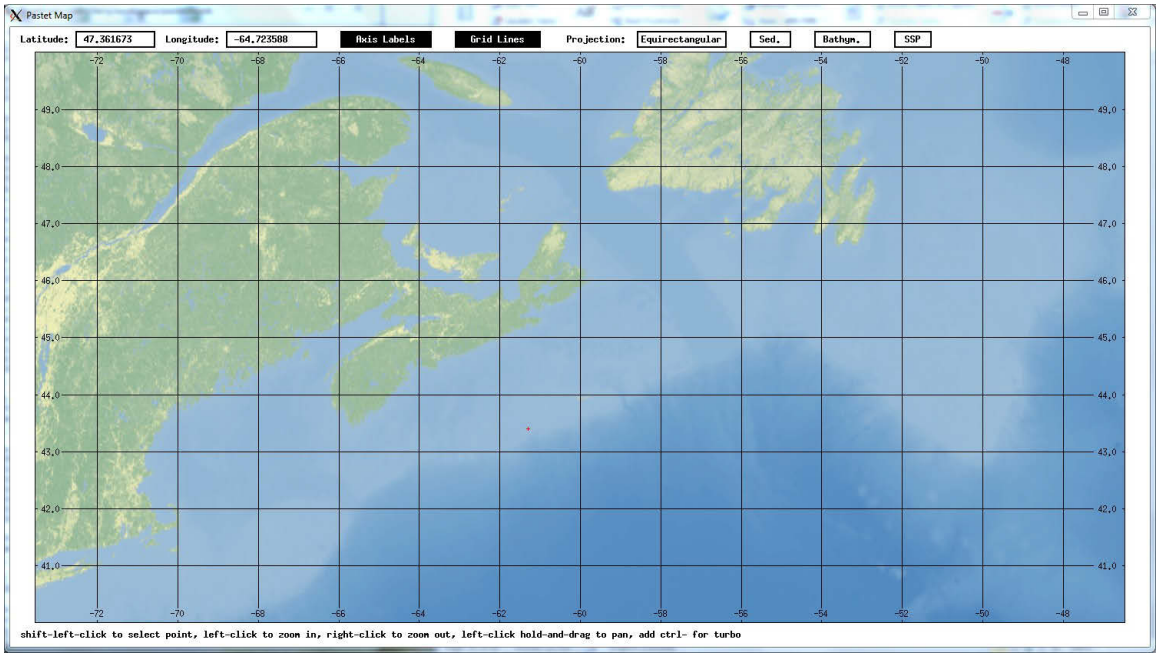

Figure 8: Example of PASTET interactive map.

The main function of the map is to allow graphical selection of the PASTET modelling location (i.e., origin point). This can be done by placing the pointer at the desired place on the map, and holding down the SHIFT key on the keyboard, then clicking the left pointer button once. When the SHIFT key is held down, the pointer changes from an arrow to a cross-hair, to allow more accurate point selection. Once the origin location point has been selected, a small red cross is placed on the map to mark the spot. The selected location point can be moved to a different place using the same procedure. The latitude-longitude of the pointer "hot spot" is continuously updated in the two text fields at the upper left of the map whenever the pointer is on the map.

The map has several other interactive functions. The axis labelling and the grid lines can be independently toggled on or off. The map projection can be selected; currently there are two map projections available: equirectangular and Mercator. Additional map projections could be added in a future upgrade of PASTET, if required.

The operator can zoom in closer to a point on the map by clicking the left pointer button; likewise, zoom-out is performed by clicking the right pointer button. The equivalent operations can also be accomplished by upward and downward rotation of the pointer device thumbwheel, if it has one. The map can also be panned in any direction by clicking the left pointer button on the

map and dragging it in the opposite direction (this has the effect of grabbing and dragging the map sheet as if it was a sheet of paper).

Both the zoom and pan functions described above can have their effects multiplied by a factor of two if the CTRL button is held down while the operation is performed.

A new feature has been added to the map with the latest PASTET release. There are now three additional options to select the display of the locations of the survey results data for sediment properties, bathymetry, and SSP, respectively. In the case of sediment properties, there are two distinct map symbols drawn: an orange diamond for sediment thickness locations and a green square for sediment type locations. The bathymetry locations are shown by a grey "x" symbol and the SSP locations are represented by a magenta "+" symbol. Figure 9 shows an example of a map where all the survey results locations are shown together, however the sediment properties, bathymetry, and SSP locations symbols can be switched on and off independently.

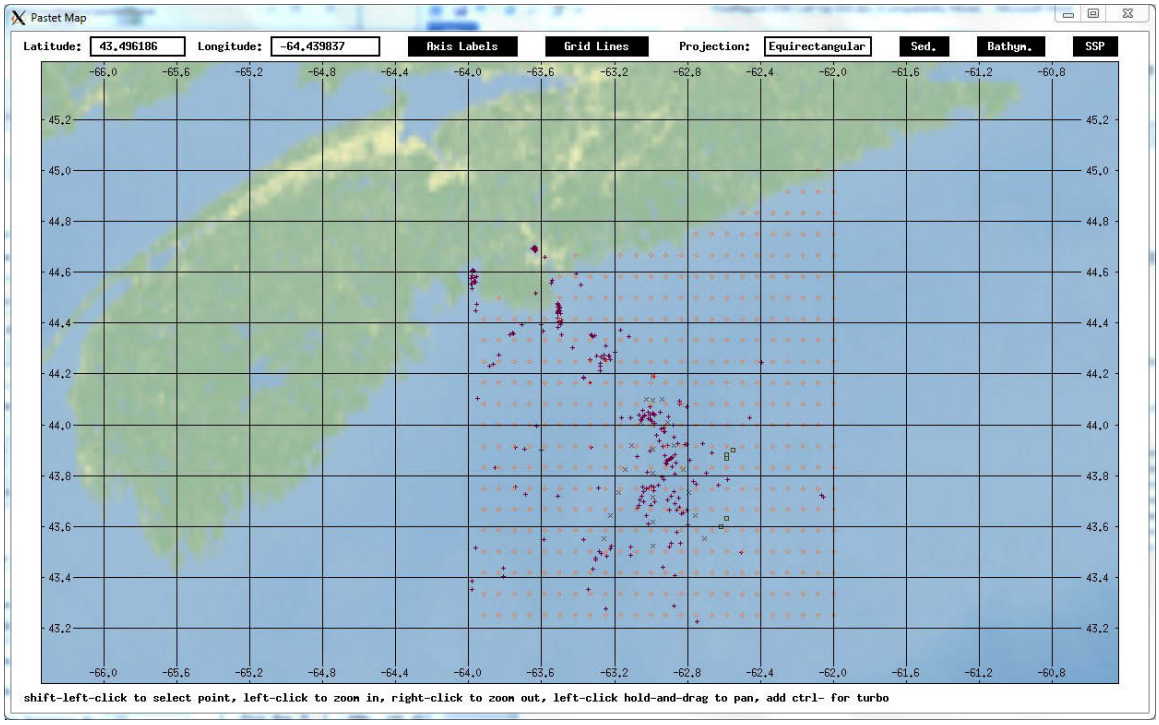

Figure 9: Example of a PASTET map display with all the optional survey results locations symbols turned on.

4 Conclusions

The PASTET program under development by DRDC Atlantic has been upgraded in several important aspects as a result of this call-up. Additional data connections to the PDB and READB databases have been added. Many software bugs have been traced and solved. The PASTET program has been installed, tested, and verified under the Linux operating system, both in 32-bit and 64-bit implementations, and under a 32-bit MS-Windows installation under Cygwin. The PASTET GUI has been updated, new plots of averaged SSPs and sensitivity versus range have been added. The interactive map has been enhanced with the optional display of survey results locations.

Some of requested tasks in this call-up could not be accomplished, due to the scope of work being bounded by a $40K fixed budget. This includes permanent access to ocean model forecast SSP data from a regularly available data source, implementation Empirical Orthogonal Functions and Monte Carlo runs for full non-linear sensitivity analysis, and two remaining output product plots.

5 Recommendations

1. Inclusion of Empirical Orthogonal Function (EOF) analysis and the capacity for Monte Carlo runs to provide full nonlinear sensitivity analysis. The investigation of this was begun in a previous call-up but there were insufficient project resources to continue implementation in recent call-ups.
2. Finish the implementation of two more output products: the full-field plot of sensitivity versus range and receiver depth, and the full-field plot of transmission loss versus range and receiver depth. These were slated for the present call-up, but project resources were exhausted before they could be completed.
3. Investigate and resolve the known unresolved bugs that are detailed in Section 2.2 above.
4. Implement the suggested enhancements that are detailed in Section 2.3 above. These are generally relatively minor changes to enhance the usability and consistency of the PASTET operations. Recommendations for more significant software enhancements are listed separately here below.
5. Implementation of additional graphical output products, including advanced display of the sensitivity results and other graphical output products that will enhance the effectiveness and utility of the PASTET application. This may include graphical display of some of the results from the environmental survey aspect of the application as well as from the propagation and sonar performance analysis aspects. One example would be a probability of detection display.
6. Enhancements to the operational and analysis capability of the PASTET application, such as support for multiple radials, uncertainty ellipsoids, etc.
7. Finish the implementation of the STB interface and complete the integration of STB data calls in PASTET. These are two discrete work items that must happen. Any technical risk associated with autonomous PASTET-STB operation is now greatly reduced, due to the progress already made under the present call-up. Finishing this interface and its integration would be an enormous benefit to PASTET operations. Most importantly, the STB sound velocity functionality should be completed for future scientific sea trials.
8. To enhance off-line scientific work using real data sourced by the STB at sea, writing the STB data to NETCDF files is recommended. These files then would be accessed through `user_exit` PDB functionality to effect off line PASTET based research using real, STB sourced data.
9. Accessing model/forecast sound velocity data in PASTET is paramount. Unfortunately, dense model data is massive and if cached in the PDB would swell the PDB data size in a similar manner that ETOPO data did with the READB. The external data approach for forecast data would work for sound velocity and other model data sources. Experience with ETOPO1 data and the PDB demonstrates reasonably fast execution with this external data approach. As well, this user-exit approach accessing NetCDF data is now relatively risk free. This approach would also permit scientists to use STB acquired data if it was also cached in a NetCDF format.
10. Once complete PDB functionality is met, the READB would be retired. This would then require the removal of READB references from PASTET.

11. Truth data and PASTET results are paramount to minimize regression in maintenance operations. PASTET is a complicated application written in an ancient paradigm and software errors are easy to make. The data bases are very sparse and finding enough locations where data is available to check the algorithmic performance is hit and miss. It is recommended that future work include a complete test set to mitigate unwanted software errors and give the scientist confidence in the contracted work.
12. PASTET currently makes limited use internally of an X11-Instrinsics facility called "work procs". When used to full effect, this will allow the X11 GUI elements to refresh and update even while a lengthy `Survey` or `Run` is in progress. The internal use of "Work Procs" in PASTET should be extended to cover those situations.
13. The databases available to PASTET for the extraction of bathymetric profiles should be extended to include commonly available stand-alone bathymetry databases, such as SRTM30plus. PASTET does not currently access those databases for the purpose of bathymetric profiles.
14. PASTET should be enhanced to use a dynamic configuration file, so that it can start up with variable values and settings that it was using when it previously shut down.
15. PASTET should be enhanced so that when the plot windows and map windows are opened, they have the same size, shape, placement, and options selected as when they were previously closed.
16. The Mercator Ocean data is available through a supplied Python script and automating the download may be possible. Recommend investigating this approach from within the DREnet for future deployments. It may be possible to progress an internal exception in the security policy for DREA Scientist to acquire this data since it is non-executable oceanographic data only.
17. Speed increases for the PDB queries may be improved through optimization and testing. Currently the speed at which some bottom and sound velocity queries execute seem slower that the READB. This is not a hard statement of fact, just an impression gained through development.
18. Once model data is firmed up, implementing a PDB strategy to cache this data would be viable. Annex E Experimental PDB Model Data Query (developed on DRDC cruise Q325) may help. This query used a table structure similar to the PDB and was hard coded for location, date and model type. However, it does work and may be used to help with the details of writing a more generic model management system for the PDB.
19. Implement a PASTET help system, especially if this tool will be used by others outside of the DRDC Modelling Group.
20. The science behind PASTET is considerable. It deals with complex physics to investigate and understand the current environmental factors of underwater sound propagation. Unfortunately the layout of the PASTET software mixes control, science, and GUI components. A much more modern design would follow the Model–view–controller paradigm.[5] At a minimum, all science (model) code should be broken out into a few C++ classes, so that portability of the PASTET science would be achieved. This concept would permit the integration of core PASTET science into other systems such as the STB/PLEIADES without control and GUI dependencies.

This page intentionally left blank.

References

[1] Call-up 010 under Standing Offer Contract #W7707-098179 Maintenance, Development, and Enhancements of the System Test Bed and Environment Modeling Manager and Associated Software.

[2] Michel Mayrand, Implementation of the Rapid Environmental Assessment Production Database, DRDC Atlantic CR 2010-123, September 2010.

[3] Mercator Ocean, http://mercator-myoceanv2.netaktiv.com/web/1-home.php.

[4] Eclipse Software Development Environment, http://www.eclipse.org/.

[5] Model–view–controller, http://en.wikipedia.org/wiki/Model%E2%80%93view%E2%80%93controller.

This page intentionally left blank.

List of symbols/abbreviations/acronyms/initialisms

ASCII	American Standard Code for Information Interchange
BP	Bottom Profile
CCW	Counter clockwise
CFAV	Canadian Forces Auxiliary Vessel
CGI	Common Gateway Interface
CNOOFS	Canadian Newfoundland Operational Ocean Forecast System
CORA	Centre for Operational Research and Analysis
CSA	Contractual Scientific Authority
CW	Clockwise
DRDC	Defence Research & Development Canada
DREnet	Defence Research Establishment Network
EMM	Environmental Modelling Manager
EOF	Empirical Orthogonal Function
ETOPO1/2/5	Electronic Topography (database), the number is the resolution in arc-minutes
FTP	File Transfer Protocol
GNU	A system of open-source, user-supported software, related to Unix (but G Not Unix)
GUI	Graphical User Interface
HTTP	Hypertext Transfer Protocol
LOB	Line of Bearing
MIME	Multipurpose Internet Mail Extensions
NCOM	Naval Coastal Ocean Model
NetCDF	Network Common Data Form
NaN	Not a Number (a flag for a floating point numerical error).
nm	Nautical Mile
PASTET	Portable Acoustic Sensitivity and Transmission Evaluation Tool
PDB	Production Database (successor to READB)
PHP	Hypertext Preprocessor
PLEIADES	DRDC Atlantic sonar platform built on the STB

PPT	Parts Per Thousand
PWGSC	Public Works and Government Services, Canada
READB	Rapid Environmental Assessment Database
SA	Scientific Authority
SE	Signal Excess
SQL	Structured Query Language
SSP	Sound Speed Profile
STB	System Test Bed
TL	Transmission Loss
TSD	Temperature Salinity Depth
XBT	Expendable Bathythermograph

Annex A Example READB SQL Query for TSD Data

```
SELECT
       temporary1.pid,
       temporary1.depth as depth,
       s.sal as salinity,
       t.temp as temperature,
       y(temporary1.the_geom) as lat,
       x(temporary1.the_geom) as lon,
       s.month as month,
       distance(temporary1.the_geom,
        geometryfromtext('POINT(43.4 -61.3)', 4269)) as dist
FROM
       tsd_salinity s,
       tsd_temperature t,
       (SELECT
              pid,
              depth,
              the_geom FROM tsd_geopoints
       WHERE
              the_geom && geometryFromText('POLYGON((-61.34999999999999 43.35,
        -61.25 43.35, -61.25 43.45, -61.34999999999999 43.45,
        -61.34999999999999 43.35))', 4269))
       AS
              temporary1
WHERE
       temporary1.pid = s.pid
       AND temporary1.pid = t.pid
       AND s.month = t.month
       AND (s.month between 1 and 1)
       AND (s.sal is not null)
       AND (t.temp is not null)
ORDER BY
       dist,
       month,
       lat,
       lon;
```

This page intentionally left blank.

Annex B Example PDB SQL Query for TSD Data

```
/*
--This sql is used to develop the symantics necessary to extract
--all salinity-depth and temperature-depth profiles within
--a time range and a two dimensional polygon.
*/
SELECT
 parameter_id AS data_type,
 data_value,
 location.depth AS depth,
 location.lat AS lat,
 location.lon AS lon,
 extract(month FROM sq.date_time) AS month,
 distance(location.geom, geometryfromtext('POINT(43.4 -61.3)', 4326)) AS dist
FROM
 scalar_quantity AS sq,
 to_date('01', 'mm') AS start_date,
 to_date('02', 'mm') AS end_date,
 (SELECT
 z(mp.geom) AS depth,
 x(mp.geom) AS lon,
 y(mp.geom) AS lat,
 mp.feature_id AS fid,
 mp.geom AS geom
 FROM
 mesh_point AS mp
 WHERE
 geom && geometryFromText('POLYGON((-61.34999999999999 43.35, -61.25 43.35, -
61.25 43.45, -61.34999999999999 43.45, -61.34999999999999 43.35))', 4326)
 AND
 mp.source='Dalhousie TSD')
AS location
WHERE
 sq.feature_id=location.fid
 AND
 to_date(cast(extract (month FROM sq.date_time) AS varchar),'mm')
 BETWEEN start_date AND end_date
 AND
 sq.data_value IS NOT NULL
ORDER BY
 parameter_id, dist, depth;
```

This page intentionally left blank.

Annex C Example READB SQL Query for XBT data

```
SELECT
       p.id_filename,
       p.depth_m as depth,
       soundspeed_mps as soundspeed,
       temperature_degc as temperature,
       y(the_geom) as lat,
       x(the_geom) as lon,
       extract(hour from time) as hour,
       extract(day from time) as day,
       extract(month from time) as month,
       extract(year from time) as year,
       distance(the_geom, point) as dist
FROM
       xbt_profiles p,
       xbt_file_meta_data m,
        to_date('01/01', 'mm/dd') as start_date,
       to_date('12/01', 'mm/dd') as end_date,
       geometryfromtext('POINT(43.4 -61.3)', 4269) as point,
       setsrid('BOX(-61.34999999999999 43.35, -61.25 43.45)'::box2d, 4269)
        as box
WHERE
       p.id_filename = m.id_filename
       AND p.depth_m > 0
       AND (temperature_degc is not null or soundspeed_mps is not null)
       AND the_geom is not null
       AND ST_contains(box, the_geom)
       AND to_date(cast(extract(day from time) || '/'
        || extract (month from time) as varchar),'dd/mm') between
       start_date and end_date
       AND extract(year from time) between 0 and 9999
       AND p.depth_m <= 1000.000000
ORDER BY
       dist,
       lat,
       lon,
       depth;
```

This page intentionally left blank.

Annex D SQL to access PDB XBT-XSV data

```
/*
The following SQL was developed to access PDB XBT-XSV data

--This sql is used to develop the symantics necessary to extract
--all sound velocity-depth and temperature-depth profiles within
--a time range and a two dimensional box given by
--(lat-long + delta/2) and (lat-long - delta/2)
*/
SELECT
 location.fid AS id_filename,
 location.depth AS depth,
 d.parameter_id AS data_type,
 d.data_value AS data_value,
 location.lat AS lat,
 location.lon AS lon,
 location.hour AS hour,
 location.day AS day,
 location.month AS month,
 location.year AS year
FROM
 data AS d,
 (SELECT
 ml.measurement_location_id AS lid,
 ip.feature_id AS fid,
 extract(hour FROM ip.date_time) AS hour,
 extract(day FROM ip.date_time) AS day,
 extract(month FROM ip.date_time) AS month,
 extract(year FROM ip.date_time) AS year,
 y(ml.geom) AS lat,
 x(ml.geom) AS lon,
 z(ml.geom) AS depth,
 distance(ml.geom, point) AS dist
 FROM
 instantaneous_point ip,
 measurement_location ml,
 to_date('01/01', 'mm/dd') AS start_date,
 to_date('12/31', 'mm/dd') AS end_date,
 geometryfromtext('POINT(43.4 -61.3)', 4326) AS point,
 setsrid('BOX(-61.4 43.3, -61.2 43.5)'::box2d, 4326) AS box
 WHERE
 ip.feature_code='XBT'
 AND ml.feature_code='XBT'
 AND ip.feature_id=ml.feature_id
 AND ST_contains(box, ml.geom)
 AND to_date(cast(extract(day FROM ip.date_time)
 || '/'
 || extract (month FROM ip.date_time) AS varchar),'dd/mm')
 BETWEEN start_date AND end_date)
 AS location
WHERE
 d.measurement_location_id=lid
       AND (d.parameter_id=322
            OR d.parameter_id=617)

ORDER BY dist, lat, lon, depth;
```

This page intentionally left blank.

Annex E Experimental PDB Model Data Query

```
Experimental PDB Model Data Query (developed on q325)

SELECT
       temporary3.pid,
       temporary3.model_run_date,
       extract(day from temporary3.forecast_date) as day,
       extract(month from temporary3.forecast_date) as month,
       extract(year from temporary3.forecast_date) as year,
       temporary3.forecast_hour as hour,
       y(temporary3.geom),
       x(temporary3.geom),
       temporary3.depth,
       temporary3.temperature,
       data_value as salinity
FROM
       model_data,
       (SELECT
              model_data.pid,
              model_data.source,
              temporary2.depth,
              temporary2.geom,
              data_value AS temperature,
              model_data.forecast_date,
              model_data.forecast_hour,
              model_data.model_run_date
       FROM
              model_data,
              (SELECT
                     model_time.pid,
                     model_time.source,
                     model_time.model_run_date,
                     model_time.forecast_date,
                     model_time.forecast_hour,
                     temporary1.depth,
                     temporary1.geom,
                     model_time.stb_data_item
              FROM model_time,
                     (SELECT
                            pid,
                            depth,
                            geom
                     FROM
                            model_grid
                     WHERE
geom && geometryFromText('POLYGON((-61.34999999999999 43.35, -61.25
       43.35, -61.25 43.45, -61.34999999999999 43.45,
                            -61.34999999999999 43.35))', 4326)) AS temporary1
              WHERE
                     temporary1.pid = model_time.pid
AND (to_date(cast(extract(day from model_time.forecast_date) || '/'
 || extract(month from model_time.forecast_date) as varchar), 'dd/ mm')
  Between to_date('01/01','mm/dd') and to_date('12/01','mm/dd'))
AND extract(year from model_time.forecast_date) between 0 and 9999
                     AND model_time.model_run_date='2009-11-03'
                     AND model_time.source LIKE 'CN04') as temporary2
```

```
        WHERE
                parameter_id = 'T'
                AND temporary2.pid = model_data.pid
                AND temporary2.source = model_data.source
                AND temporary2.model_run_date = model_data.model_run_date
                AND temporary2.forecast_date = model_data.forecast_date
        AND temporary2.forecast_hour = model_data.forecast_hour) AS temporary3
WHERE
        parameter_id = 'S'
        AND temporary3.pid = model_data.pid
        AND temporary3.source = model_data.source
        AND temporary3.forecast_date = model_data.forecast_date
        AND temporary3.forecast_hour = model_data.forecast_hour
        AND temporary3.model_run_date = model_data.model_run_date;
```

Annex F Pastet and STB Installation

```
Pastet and STB Installation Ubuntu 10.04

OS Installation

Install Ubuntu 10.04 LTS

Set Proxy
       menu System/Preferences/Network Proxy
       select Manual proxy configuration
       select Use same proxy for all protocols
       HTTP Proxy: webproxy.drdc-rddc.gc.ca
       Port: 8080

Update operating system
       menu System/Administration/Update Manager
       select Check
       when package information is downloaded
       select Install Updates

Prepare for Pastet and the STB

       menu System/Administration/Synaptic Package Manager
       using the Quick search
       enter
              build-essential
                     mark for installation...apply
              tcsh
                     mark for installation...apply
              libpng-dev
                     select libpng12-dev
                     mark for installation...apply
              libx11-dev
                     mark for installation...apply
              gfortran
                     mark for installation...apply
              bzr
                     mark for installation...apply
              python-paramiko
                     mark for installation...apply
                     Note: this may have been done with bzr
              postgres-client
                     select postgres-client-common
                     mark for installation...apply
              pgadmin3
                     mark for installation...apply
              libpq-dev
                     mark for installation...apply
              libxt-dev
                     mark for installation...apply
              libxaw7-dev
                     mark for installation...apply
              libgeotiff-dev
                     mark for installation...apply
              libtiff4-dev
                     mark for installation...apply
```

```
Pastet Installation

In Pastet directory edit the dbAccess.cfg file and file in the setting for the
ldb stb and pdb. Recommend connecting to the databases using pgadmin3 prior to
editing this file.

Installing Postgres Server for Deployments and Development purposes

       menu System/Administration/Synaptic Package Manager
       using the Quick search
       enter postgres
              postgresql-doc-8.4
                     mark for installation...apply
              postgresql-doc
                     mark for installation...apply
              postgresql
                     mark for installation...apply

Setup the postgres system
              to start psql as the postgres default user
              sudo -u postgres psql template1
       change the psql user(defualt user password)
              \password postgres
              enter postgres (twice)
              cntl d to exit

To start psql client as the postgres default user
       sudo -u postgres psql

Modify /etc/postgresql/8.4/main/pg_hba.conf file to set required permissions

For a local system the following lines would suffice (however ensure your
systems IT team is aware if outside network access is required)

# Database administrative login by UNIX sockets
local   all         postgres                            trust

# TYPE  DATABASE    USER        CIDR-ADDRESS          METHOD

# "local" is for Unix domain socket connections only
host   all        all         131.136.0.0/16         md5
local all         postgres                           trust
# IPv4 local connections:
host    all         all         127.0.0.1/32          trust
# IPv6 local connections:
host    all         all         ::1/128               md5

To restart the postgres server
       sudo /etc/init.d/postgresql-8.4 restart

Default directories used by Postgresql

       help/man pages                          /usr/share/postgresql/8.4
       user_exit files (CDF)                   /usr/share/postgresql-common
       server and utility postgresql programs  /usr/lib/postgresql/8.4
       startup and control scripts             /etc/postgresql/8.4/main
       restart, stop, start server             /etc/init.d/postgresql-8.4
```

```
Install Postgis
       menu System/Administration/Synaptic Package Manager
       using the Quick search
       enter postgis
              postgresql-8.4-postgis
              mark for installation...apply

Installing the readb from a backup database
       From the command line create the database
              sudo -U postgres createdb readb
       Load the backup backup.sql
              sudo -U postgres psql -d readb -f  backup.sql

Installing the pdb from a backup database
       Similarly load the pdb from the backup (assuming that the postgres
        user has local trust assigned earlier) the following alternate
      syntax should also work:
       psql -h 127.0.0.1 -U postgres pdb < pdb.sql



User-Exit Setup
       menu System/Administration/Synaptic Package Manager
       using the Quick search
       postgresql-plperl
              postgresql-plperl-8.4
              mark for installation...apply

       Ensure Java is installed and recommend sun-java-6
       menu System/Administration/Synaptic Package Manager
       using the Quick search
              sun-java6-jdk
                     sun-java6-bin
                     sun-java6-jre
                     sun-java6-jdk
                     mark for installation...apply

Tell the pdb to accept the plperl untrusted language.
       From a command window:
              psql -U postgres -h 127.0.0.1 -d pdb
              CREATE LANGUAGE plperlu;
```

This page intentionally left blank.

Annex G geo_acoustic_interface

```
/*
 * geo_acoustic_interface.h
 *
 *  Created on: 2011-03-02
 *      Author: fred campaigne
 */

#ifndef GEO_ACOUSTIC_INTERFACE_H_
#define GEO_ACOUSTIC_INTERFACE_H_
/*
 * geo_acoustic.h
 *
 *
 *
 *  1.   The SI system is used except:
 *          a. Degrees for angular measure
 *          b. Degrees Celsius for temperature measurements.
 *          c. Practical Salinity Scale (PSS) for salinity values.
 *       Longitude and latitude are in decimal degree format.
 *
 *  Results are returned by reference in the last argument and a reference to a
 *  constant will be passed in the second last argument to indicate a missing or
 *  unknown value.
 *
 *  2.   The stb will re-size the std based vectors as required.
 *       Depth values use the std::pair format, for example:
 *
 *       temperature.first is the depth in meters and temperature.second
 *       is the temperature in degrees celsius.
 *
 * 3.   The following vectors are used:
 *
 *       a single sound velocity, temperature or salinity point,
 *       a one dimensional vector of depth pair values are used
 *       (depth, temperature) or (depth, salinity) or
 *       (depth, sound velocity).
 *
 *       For bearing lines depth is a one dimensional vector of depths, that is
 *       nPoints long and spaced by the resolution in meters.  For salinity,
 *       temperature and sound velocity is a nPoints long one dimensional
 *       vector of one dimensional vectors of depth value measurements. For
 *       example a point (depth temperature pair) in a line of salinity would
 *       be salinity[line_index][depth_index]
 *
 *       Grid depth values are contained in a two dimensional vector of depth
 *       values indexed by longitude and latitude (Depth[lon_index][lat_index]).
 *       Temperature, salinity and sound velocity will be a two dimensional
 *       vector of one dimensional vectors of depth value pairs
 *       (Salinity[lon_index][lat_index][depth_index])
 *
 *  4.   Time may or may not be used by the called routine and will be the Unix
 *       standard cast to a double for fractional second accuracy.
 *
 *  5.   Bearing lines will follow the great circle.
 *
```

```
 *  6.   All grid based interfaces are referenced relative to grid centre with
 *       the origin (0, 0) at the top left corner.
 *
 *  7.   All grid based interfaces will use the same resolution on both latitude
 *       and longitude.
 *
 *  8.   The return indicates success or failure.  A query that was filled with
 *       missing values is a success.
 *
 */

#include <vector>
#include <utility>

namespace ga_data
{

class geo_acoustic_interface
{

public://types
   typedef std::vector<double> vector;
   typedef std::vector<std::vector<double> > grid_depth_vector;
   typedef std::pair<double, double> depth_and_value;
   typedef std::vector<depth_and_value> depth_value_vector;
   typedef std::vector<depth_value_vector> line_value_vector;
   typedef std::vector<std::vector<depth_value_vector> > grid_value_vector;

public:
   /**
    * province number between 1 and 7
   */
   virtual int bottom_type(double time,
                  double lon,
                  double lat)const = 0;

   //not expected initially
   virtual int bottom_type(const char* file) = 0;

   //single point interface
   virtual bool depth(double time,
              double lon,
              double lat,
              const double& missing,
              double& result)const = 0;

   virtual bool temperature(double time,
                    double lon,
                    double lat,
                    const double& missing,
                    depth_value_vector& result)const = 0;

   virtual  bool salinity(double time,
                 double lon,
                 double lat,
                 const double& missing,
                 depth_value_vector& result)const = 0;

   virtual bool sound_speed(double time,
                    double lon,
```

```
                double lat,
                const double& missing,
                depth_value_vector& result)const = 0;

/*
 * line of bearing interface
 * Starting from the latitude and longitude parameters following the
 * great circle for nPoints at the given distance resolution.
 */

virtual bool depth(double time,
           double lon,
           double lat,
           double bearing,//in degrees
           double resolution,//meters
           long nPoints,
           const double& missing,
          vector& result)const = 0;

virtual bool temperature(double time,
                double lon,
                double lat,
                double bearing,//in degrees
                double resolution,//meters
                long nPoints,
                const double& missing,
                line_value_vector& result)const = 0;

virtual bool salinity(double time,
              double lat,
              double lon,
              double bearing,//in degrees
              double resolution,//meters
              long nPoints,
              const double& missing,
              line_value_vector& result)const = 0;

virtual bool sound_speed(double time,
                double lat,
                double lon,
                double bearing,//in degrees
                double resolution,//meters
                long nPoints,
                const double& missing,
                line_value_vector& result)const = 0;

/*
 * Grid interface
 * The latitude and longitude define the centre of the grid. Both
 * directions extend for an equal number of points at the same
 * distance resolution.
 */

virtual bool depth(double time,
           double lon,
           double lat,
           double bearing,//in degrees
           double resolution,//meters
           long nPoints,
           const double& missing,
```

```
            grid_depth_vector& result)const = 0;

   virtual bool temperature(double time,
                  double lon,
                  double lat,
                  double bearing,//in degrees
                  double resolution,//meters
                  long nPoints,
                  const double& missing,
                  grid_value_vector& result)const = 0;

   virtual bool salinity(double time,
                double lat,
                double lon,
                double bearing,//in degrees
                double resolution,//meters
                long nPoints,
                const double& missing,
                grid_value_vector& result)const = 0;

   virtual bool sound_speed(double time,
                  double lat,
                  double lon,
                  double bearing,//in degrees
                  double resolution,//meters
                  long nPoints,
                  const double& missing,
                  grid_value_vector& result)const = 0;


};

}//namespace

#endif /* GEO_ACOUSTIC_INTERFACE_H_ */
```

DOCUMENT CONTROL DATA

(Security classification of title, body of abstract and indexing annotation must be entered when the overall document is classified)

1. ORIGINATOR (The name and address of the organization preparing the document. Organizations for whom the document was prepared, e.g. Centre sponsoring a contractor's report, or tasking agency, are entered in section 8.)

JASCO Applied Sciences
32 Troop Ave,
Dartmouth, NS Canada B3B 1Z1

2. SECURITY CLASSIFICATION (Overall security classification of the document including special warning terms if applicable.)

UNCLASSIFIED
(NON-CONTROLLED GOODS)
DMC A
REVIEW: GCEC APRIL 2011

3. TITLE (The complete document title as indicated on the title page. Its classification should be indicated by the appropriate abbreviation (S, C or U) in parentheses after the title.)

PASTET Development - Use of Sound Speed Profiles

4. AUTHORS (last name, followed by initials – ranks, titles, etc. not to be used)

Deveau, T.J.; Campaigne, F.; Widdis, E.

5. DATE OF PUBLICATION (Month and year of publication of document.)	6a. NO. OF PAGES (Total containing information, including Annexes, Appendices, etc.)	6b. NO. OF REFS (Total cited in document.)
July 2012	58	5

7. DESCRIPTIVE NOTES (The category of the document, e.g. technical report, technical note or memorandum. If appropriate, enter the type of report, e.g. interim, progress, summary, annual or final. Give the inclusive dates when a specific reporting period is covered.)

Contract Report

8. SPONSORING ACTIVITY (The name of the department project office or laboratory sponsoring the research and development – include address.)

Defence R&D Canada – Atlantic
9 Grove Street
P.O. Box 1012
Dartmouth, Nova Scotia B2Y 3Z7

9a. PROJECT OR GRANT NO. (If appropriate, the applicable research and development project or grant number under which the document was written. Please specify whether project or grant.)	9b. CONTRACT NO. (If appropriate, the applicable number under which the document was written.)
	W7707-098179/001/HAL

10a. ORIGINATOR'S DOCUMENT NUMBER (The official document number by which the document is identified by the originating activity. This number must be unique to this document.)	10b. OTHER DOCUMENT NO(s). (Any other numbers which may be assigned this document either by the originator or by the sponsor.)
DRDC Atlantic CR 2012-101	

11. DOCUMENT AVAILABILITY (Any limitations on further dissemination of the document, other than those imposed by security classification.)

Unlimited

12. DOCUMENT ANNOUNCEMENT (Any limitation to the bibliographic announcement of this document. This will normally correspond to the Document Availability (11). However, where further distribution (beyond the audience specified in (11) is possible, a wider announcement audience may be selected.))

Unlimited

13. ABSTRACT (A brief and factual summary of the document. It may also appear elsewhere in the body of the document itself. It is highly desirable that the abstract of classified documents be unclassified. Each paragraph of the abstract shall begin with an indication of the security classification of the information in the paragraph (unless the document itself is unclassified) represented as (S), (C), (R), or (U). It is not necessary to include here abstracts in both official languages unless the text is bilingual.)

DRDC Atlantic has been developing a research-level acoustic prediction system, the System Test Bed (STB), in support of tactical decision aids and improving operator effectiveness. The Environmental Modeling Manager (EMM) is one of the STB components. The Portable Acoustic Sensitivity and Transmission Estimation Tool (PASTET) has been, and is being, developed as an adjunct to the STB and extension of existing acoustic propagation modeling capabilities. The first objective of this call-up is to include the capability for using mean and standard deviation of sound speed profiles in PASTET, along with several bug fixes. The second objective of this call-up is to provide the scientific authority with the tools required to use ocean forecast models as inputs to EMM and PASTET. This report describes the work that was done during the call-up, the issues encountered, the results that were produced, and includes recommendations for additional work.

RDDC Atlantique a mis au point un système de prédiction acoustique de recherche, le Banc d'essai de systèmes (*System Test Bed,* STB), dans le but d'appuyer le développement d'aides à la prise de décisions tactiques et l'amélioration de l'efficacité de l'opérateur. L'EMM (Environment Modeling Manager – *gestionnaire de modélisation de l'environnement*) est un des éléments du STB. PASTET (Portable Acoustic Sensitivity and Transmission Estimation Tool – *Outil portable d'évaluation de la transmission acoustique et de sa sensibilité*) est un logiciel mis au point pour servir d'appui au STB et étendre les capacités existantes de modélisation de la propagation acoustique dont le développement se poursuit. Le premier objectif de la présente commande est de permettre l'utilisation de la moyenne et de l'écart type des profils de vitesse du son dans PASTET et de régler plusieurs bogues. Le deuxième objectif est de fournir à l'autorité scientifique les outils nécessaires pour utiliser des modèles de prévision océanique pour fournir des données d'entrée à l'EMM et à PASTET. Le présent rapport décrit les travaux effectués pour cette commande, les problèmes survenus et les résultats produits. Il comprend également des recommandations sur des travaux supplémentaires.

14. KEYWORDS, DESCRIPTORS or IDENTIFIERS (Technically meaningful terms or short phrases that characterize a document and could be helpful in cataloguing the document. They should be selected so that no security classification is required. Identifiers, such as equipment model designation, trade name, military project code name, geographic location may also be included. If possible keywords should be selected from a published thesaurus, e.g. Thesaurus of Engineering and Scientific Terms (TEST) and that thesaurus identified. If it is not possible to select indexing terms which are Unclassified, the classification of each should be indicated as with the title.)

Acoustic propagation modeling, sensitivity analysis

This page intentionally left blank.

Defence R&D Canada

Canada's leader in defence
and National Security
Science and Technology

R & D pour la défense Canada

Chef de file au Canada en matière
de science et de technologie pour
la défense et la sécurité nationale

DEFENCE DÉFENSE

www.drdc-rddc.gc.ca