Cook Islands Ministry of Internal Affairs
Division of Social Welfare
Final Report

Student Consultant, Tim Garvey

Community Partner, Ngatuaine Maui

I. About the Ministry of Internal Affairs

Organization

Taking into account the 2011-2015 National Sustainable Development Plan and the 2012-13 Budget Policy Statement targeting the two goals of improving the social wellbeing of Cook Islanders and supporting economic enablers through regulation of employer and business behavior, the 2012-13 vision is:

> *Opportunity for all People who reside in the Cook Islands and A Vibrant Cook Islands Economy.*

The Ministry of Internal Affairs (INTAFF) and all of its divisions are located along the main road in Tupapa. The Division of Social Welfare (DSW) is a federal office given powers and responsibilities by legislation or Cabinet ruling to provide for the social wellbeing of the Cook Islanders. The major task of the DSW is to make payments to eligible beneficiaries in a timely manner through effective monitoring systems. Managing a Social Impact Fund and increasing care for the elderly will play a larger role as the mission of the DSW responds to circumstances.

Forty-two employees work throughout INTAFF; within the DSW, five full-time staff in Rarotonga and an additional welfare officer for each of the outer islands (9) complete the work. In 2010-11, INTAFF was responsible for the administration of nearly NZ$12 million dollars of Crown funding for welfare payments to beneficiaries. Records show that for the financial year 2010-11, welfare assistance and support was provided to 5,699 people who received a payment through a pension, child benefit, destitute or infirm relief, caregiver's allowance or other subsidy. The operational budget for 2011-2012 was NZ$12,000.[1] These amounts are expected to alter significantly in the future as the population simultaneously ages and decreases.

A 2011 outside audit consistently sites the lack of quantitative data for monitoring, evaluation, forecasting, and other analysis. Other problems include lack of a mechanism to learn of beneficiaries leaving the country in order to cease payments and other checks against welfare fraud such as multiple names, concurrently collecting benefits from other countries, and types of benefits received (i.e. both caregiver and an infirmed benefit) without manually checking each specific case. Currently, the DSW depends on the community for such abuses and the duties to check for these are delegated to the welfare officers, who are occupied with benefit processing.

INTAFF has no in-house ICT staff and any breakdown in technology can be down for weeks. As identified in the audit, there are large gaps in the policy and legal framework for the Ministry.

[1] 2011 Performance Review of the administration of welfare payments by INTAFF

Furthermore, the Ministry is too under-resourced to prevent things like fraud and abuse. Like other island nations, they face large limitations in terms of bandwidth and local technical capacity.

Facilities

INTAFF is located east of Avarua in Tupapa, but will likely move to a nearby location in one year when the current lease is up. The aged building is one story and lies directly between the main road and the seaside, has a somewhat piecemealed appearance due to the many divisions and additions over the years, and most of the building is concrete with some portions consisting of wood paneling; the roof leaks during inclement weather. The DSW lies in the middle of the building behind a small courtyard where the public enters and consists of three rooms: the main reception area where the two Welfare Officers work, a shared conference room, and the Director's office. The Division of Censorship's office can only be accessed through the DSW, but many of the other divisions regularly visit the space.

The building is powered with lighting and fans throughout, as well as one security light outside; a security guard monitors the building from 9 pm to 6 am. Louvre windows access every room and the DSW was broken into last year, though nothing was taken.

The space in DSW is frequently used by INTAFF employees, though the public visits about three times a day and other divisions to a lesser extent. Each employee uses a computer at their desk with network connections. The computers have been networked to one main printer. Network and mail servers for the Ministry are housed within.

Programs

INTAFF is somewhat of a catchall for all things domestic for the Cook Islands. Divisions include Social Welfare, Public Service, Censorship, Child and Family, Disability, Gender, Labor and Consumer, Finance, amongst others. Temporary workers from the UN and other organizations frequently work within different divisions within INTAFF.

The DSW runs several social programs for the country. The two main programs are the Old Age Pension and the Child Benefit Allowance, in which financial assistance is given to eligible individuals over the age of 60 or with children under the age of 12, respectively. Other programs include the new born allowance, infirm, destitute, caregiver, funeral allowance, and other payments; the amount from these programs depends on time in country. All payments from these programs go through the Ministry of Finance and Economic Management (M-FEM).

The DSW also manages a new Social Impact Fund, which provides funding to local non-profit and community organizations' projects and programs based on their activities relating to focus areas such as gender equality, youth empowerment, disabilities, the elderly and mental health. .

Staff

The DSW contains three staff members for welfare and two staff for other programs; a consultant is also associated with the DSW. The Secretary oversees all divisions within INTAFF. All use Microsoft Word 2007 and Excel for communication, creating reports, and tracking applicants through the process; Outlook is used for email. Internet Explorer is the browser of choice for internet browsing. No training has been provided; the internet is often consulted for technical issues.

Junior Welfare Officer – Iva Eitiare: Processes applicant claims, checks Senior Welfare Officer's processing, creates top sheet summaries for applicants

Senior Welfare Officer – Helina Glassie: Processes claims, checks Junior Welfare Officer's processing, maintains current Welfare Excel database. Considered most experienced with technology and has been in the INTAFF the longest and troubleshoots problems

Welfare Director – Ngatuaine Maui: Checks outgoing payments prior to secretary approval, generates reports and keeps track of budget, oversees divisional communications.

Social Impact Fund Director – Angie Tuara: Would like to keep track of organizations, their projects, and metrics for assessing impact, but the program design is still in the early phases.

Senior Inspector - (Auntie) Nga: Mostly emails outside parties. She might need a system for case management of the elderly in the future.

Consultant – Debbie Ave: Completed review of DSW. Has funding to administer to NGOs for nursing and homecare, and could benefit from a system similar to the Social Impact Fund for evaluating effectiveness of funding.

Secretary of Internal Affairs – Bredina Drollet: Uses Excel for budgeting and trend analysis. Uses other applications for communication

Technology Infrastructure

Technology	Description
Hardware	An IBM x3200 network server for the Ministry with Intel Core 2 Duo, E7200 @ 2.53 GHz, 3 GB RAM, 900 GB hard disk with 400 GB free and a desktop running the mail server are housed within INTAFF with UPS power backup The database server has Intel Core i5 2400 3.10GHz, 8GB DDR3 RAM, 1TB HDD, DVD Writer, Gigabyte GV-N210D3-1GI 1GB PCIE HDMI+DVI+VGA Graphics Card, and Windows 7 (32 bit) Desktops within the DSW are refurbished Dell Optiplex GX620's with Pentium D CPUs, 3.00/2.99 GHz, 1 GB RAM, floppy, CD, sound, Ethernet, 40GB hard drive, and 8 USB ports
OS	Desktops: Windows XP Professional, SP3 Server: Windows 2003 Server
Peripherals	Flat screen monitors and keyboard/mouse that connect via USB, one Brother HL-2140 laser printer
Software	MS Office 2007 is the main software used throughout the office with Word, Excel, and Outlook being the main programs. Labor and Consumer Affairs uses an Access database and Child and Family Services uses a MySQL database, but cannot be administered on the current server. Other divisions (including Social Welfare) maintain Excel databases as needed.
Network	A small WiFi router is connected to the server, though this was supposed to have been disabled and does not have a strong enough signal for access by most of the Ministry; the Public Service Commission (housed in a building behind the Ministry) connects to the server via a different

	WiFi connection. Other workstations have a traditional network, though much of the hubs and other equipment are over five years old; a main printer is included in this network, which malfunctions often, so most users use divisional desktop printers.
Internet Connection	All stations are connected to an intranet with access to the internet
Other	The conference room contains approximately 20+ defunct computers, as well as keyboards, that were donated from New Zealand, but have been assessed by the Ministry as inoperable and are used for spare parts.

Issues: For most day-to-day tasks, this level of technology is sufficient; however, more ambitious plans for forecasting, budgeting, and preventing abuse of the system will eventually require equipment upgrades if done internally. Thus, a new machine will be ordered.

Technology Management

Technological management is supposed to be centralized at some point in the future by the ICT division within the Office of the Prime Minister (OPM); however, in practice, they are currently only providing technical training and oversight of communications (websites) directly to ministries. They are also supposed to be coordinating a central offsite backup location, but backups are being done and held onsite. Technology management is done independently by INTAFF and comes out of operating budgets.

Technology infrastructure is managed by an outside contractor, The Computer Man, who also handles major troubleshooting, but there is often a large delay between call and response. A log is maintained of all problems. Smaller, Windows-related, issues are often handled by the Senior Welfare Officer. Each division is responsible for the maintenance of their own machines, including the server, and anti-virus and software updating is done via the network. Software upgrades are limited by hardware, which is limited by other resources.

Technology Planning

No technology planning currently done and is dependent on funds made available in the federal budget. Otherwise, ministries often receive second-hand donations from New Zealand, which is of differing quality. When funds are made available, each division is responsible for their own allocation. The DSW meets as a committee to decide what is needed, where the Senior Welfare Officer has heavy influence due to experience. The DSW recently decided to purchase a computer for use as a database server and tool for development for the Ministry; this will meet most of the recommended requirements for running the current version of MySQL Workbench.

ICT within the OPM will supposedly transfer all technology over to the LAMP (Linux, Apache, MySQL, PHP) stack. They are currently working on establishing a quality database in most divisions and ministries that will eventually be merged into a central system and plan on offering training and advanced database analysis. However, all these plans are currently in very early stages and subject to change over several years.

Problems with technology ministry-wide are normally desktop issues such as locating a file or difficulty saving or editing and other issues due to not knowing how the software functions on an underlying level; in these cases, the Senior Welfare Officer can normally find the solution. Other common problems are server outages in which the outside contractor is called. Other welfare programs around the world have sophisticated systems to check identity (fingerprint, etc.) and other fraud preventions as well as other, more centralized, systems that are not currently in place by the DSW. Likewise, the current system is inadequate for tracking, forecasting, or monitoring on most levels.

Internal Communication

An intranet exists in which each division can access printers and their own files with a shared *corporate* directory. Each staff member has their own email account and internet access that is capped at 6.75 GB/month for the ministry before overage charges are incurred; individuals do not have maximums but several websites are blocked. Inter-office communication is done in person or via email depending on the nature of the communication; files are shared over the network or by flash drive for those few stations not on the network. No voicemail currently exists and office lines are unable to call mobile phones. Communication by outer island welfare officers is currently done by phone or fax; though, the Ministry is currently working on refining this method where the outer island welfare officers will use the administrative offices on each island (at that office's cost), where they will have internet access and be able to send forms in that manner. Contact information of beneficiaries is housed in an Excel database with more robust information remaining on the beneficiary's application.

An outdated website exists with basic information on the responsibilities of each division and how to contact them; however, this website has not been updated in years and the Ministry is restructured often. Due to the outdated information on the website and the nature of communication within the Cook Islands, most communication with applicants is done over the phone or in person. External communication via YouTube or other common websites may increase awareness of programs for those Cook Islanders living abroad, which may help facilitate the current policy of combatting depopulation; domestically, these mediums would be ineffective due to the slow and costly internet access nationwide.

The DSW currently has only one application form for all benefits, which was recently revised to simplify application and processing. They have identified that the processing portion of this form is confusing and the Junior Welfare Officer makes a top-sheet in lieu of this portion of the application; this method is yet to be endorsed by the Director and may be obsolete as new systems take effect. After processing and approval, the applicant of said application may receive an "additional voucher number" that goes with the amount due to the beneficiary until the next disbursement cycle; it will also receive a "main voucher number" for the amount due in the next disbursement cycle. These amounts are checked by the Director, the Finance Division, and finally the Secretary before being forwarded to the M-FEM for payment.

Information Management

Critical information includes beneficiary information, particularly their arrears, and bank information, and accounting information. Beneficiary information is managed by an Excel spreadsheet containing some information from the paper application and a separate spreadsheet for disbursements. Voucher numbers are generated by the DSW, but accounting information is handled electronically by the Finance Division, DSW brings hardcopies to the Division of Finance, where they are photocopied and signed off by that division, DSW then keeps a copy on file. Both divisions also keep paper records. Critical information from the paper application is entered into the Excel database and even more critical information is then entered on the weekly processing forms and top sheet of each application.

No information system is in place to manage workflow or increase efficiency. A student consultant assisted in the creation of Access databases in 2005, which are still on file, but these have been abandoned due to changes in the process, policy, and development capacity to alter these sufficiently. The DSW would like to automate as much of the processing as possible, as well as check with the Immigration, Justice, and M-FEM. Current systems make it difficult to detect fraud and there is no system with Immigration to confirm dates of departure, Justice to confirm identities, or Finance to speed up payments; no automated internal checks are in place to detect abuse. Some work is duplicated by entering the paper form into the database and copying relevant information to subsequent steps.

The current database was built internally by staff with Excel and another database exists in the Division of Disability Services. A MySQL database was built by an expert from New Zealand for the Division of Child and Family Services and is being tested and maintained by a consultant from another ministry. Consumer and Labor Services built their own Access database that they have had trouble with since upgrading from Access 2003 to Access 2007. The current database in DSW does not offer the ability to run queries that offer insight for management. Furthermore, current systems do not offer a monitoring method of DSW's ability to provide timely welfare payments within an effective monitoring system.

Business Systems

Accounting is done by the Division of Finance for the Ministry. They handle welfare payments as well as payroll with Manage Your Own Business (MYOB) 10, but keep track of employee benefits in Excel, as advised per a 2009 audit that stated it was not worth the trouble to track benefits in MYOB. Finance reports that the current system does not have problems, but the Secretary stated that there are sometimes problems with reconciliation of welfare payments. If monitoring of the programs and steps to prevent abuse were built into the system, donor agencies would be more likely to fund programs and initiatives.

II. Data Management System of Welfare Beneficiaries

Motivation

DSW's system for collecting beneficiary data consists of manual entry into a spreadsheet, which is then used to produce weekly voucher payments for the Accounting Division. While this is adequate for processing of payments, it fails to meet all the needs of the DSW:

- ❖ Does not allow for forecasting, tracking, or monitoring of programs
- ❖ Does not have systems in place to check for program abuses
- ❖ Requires Welfare Officer time and focus
- ❖ Cannot be easily integrated with current and future systems in other ministries
- ❖ Relies on outdated software that will be phased out in coming years (Office 2003)

The inability to forecast and analyze trends leads to uninformed and poor planning by management. This has been stated in interviews with both the Director of Welfare and the Secretary of Internal Affairs as well as outside audits that also site the lack of monitoring allowed by the current system.

Welfare abuse can take many forms: multiple names, conflicting benefits (both infirm and caregiver), false time in country and other forms of dishonesty. Since the DSW does not currently have the capacity to investigate abuse directly, a robust system is crucial. The current system does not have any checks in place and abuses can only be discovered by directly investigating individual beneficiaries when abuse is rumored. Abuse results in payments due back to the DSW by the beneficiary, which requires large amounts of administrative work, financial pressure on the beneficiary, and potential budgeting issues for the DSW.

As there is no automation in the current system, Welfare Officers do the work manually. Consisting of approximately 2-5 new entries or adjustments daily, 1-2 reports a week, and summaries for over 5,000 benefits, this takes time and is more prone to error than a more appropriate system.

The Cook Islands government has been consistently moving towards more centralization, particularly in terms of technology. The current system does not lend itself to centralization or integration with other systems and is not consistent with OPM's movement towards a LAMP framework. Integration between systems would increase efficiencies for all ministries. OPM is moving toward a LAMP framework and away from Microsoft products. Microsoft products are more costly, more prone to attack, and do not offer the same levels of functionality as more sophisticated products. As these systems are phased out, the current system will become obsolete.

Many options exist for a system to manage beneficiary data, from paper based, cloud computing, an off-the-shelf system, or a custom system. Government is trying to move away from paper based systems to improve efficiency and accuracy. Cloud-based systems will be too costly, given bandwidth costs. Hewlitt-Packard and Accenture, two firms offering systems for managing welfare benefits we contacted as to what they could provide the DSW, given the constraints of remote location, costs, and time-frame; Hewlitt-Packard did not respond and Accenture formally declined to provide services. Using two open-source solutions, LibreOffice's Base (similar to Access) and MySQL (similar to Oracle), offers the support provided by open source communities, is free, and can be customized to the needs of DSW. Moreover, it allows a consistent platform in which to integrate with other ministries and allows OPM the ability to offer assistance and training to the ministries in the future.

Outcomes

A Process Overview, based on interviews, observation, documents, and systems in place was created as a guide in system design. From this overview, the required data was assessed and normalized (or "segregated"), and placed in an entity-relationship diagram. The diagram can be observed under the *Data Modeling* section of MySQL Workbench on the DSW server (and by looking at *Relationships* in Base) and the normalized data can be found in the database.

The database was implemented within the DSW as follows:

- MySQL Connectors, Java 6, and Visual C++ Redistributable 2010 were installed on the Welfare Officer's stations, but not on the Welfare Director's station, as this was out of the office for the last two weeks of the consultant's term at MINTAFF
- LibreOffice 3.5.5's Base, Writer, and Calc were installed and set-up on the Welfare Officer's stations
- A server was purchased to act as a database server for the entire Ministry of Internal Affairs

- Windows 7, Java 6 and 7, Visual C++ 2010 Redistributable, MySQL Server, MySQL Workbench, and LibreOffice 3.5.5 were installed onto the server
- The server was networked into the MINTAFF office network
- A database, *welfare*, was created on the server.
- An .odb Base file was created as a client-end system for inputting data and generating vouchers. The final version Welfare_System_2012a.odb is with the Senior Welfare Officer for migration to other stations
- Data from vouchers from the outer islands was imported into the database and the Senior Welfare Officer input data using the *Beneficiary Application* form
- The Base file generates several reports with parameters used by DSW:

  However, several reports such as Christmas bonuses, taxes, more specialized benefits, and managerial reports were not created.
- The Senior Welfare Officer was trained in inputting data, editing forms, creating reports, editing reports and how they interact with the database, and basics of database administration using MySQL Workbench. In the process, the officer edited a form, input data, created reports with dummy data, and generated a user account for the Junior Welfare Officer.
- An 11 page manual, Welfare_System_Manual.doc, was written advising on client-end use, basic MySQL tasks, common pitfalls, MySQL Workbench, and additional help

The database itself has seven tables and two views.

Base interacts with the database through one form for inputting data. It also has eight queries with parameters for extracting information from the views:

- One query for Additional Vouchers (including the Newborn Allowance)
- Three Main Voucher queries for the Outer Islands based on Old Age Pensions, Child Benefit Allowances and Caregiver/Infirm/Destitute Allowances
- Three Main Voucher queries for Rarotonga based on Old Age Pensions, Child Benefit Allowances and Caregiver/Infirm/Destitute Allowances
- One general query for Main Vouchers for any benefit

Base also has eight reports based on each of these queries. However, the general report is for reference and does not need to be used. However, the system does not track changes over time, as is currently done in the voucher summaries and this will still have to be done through a spreadsheet.

The bulk of the training focused on the Senior Welfare Officer and, thus, depends heavily on their presence and ability to train the rest of the DSW. Likewise, most of the training was spent on the client-end, as this component will be used the most, but may create problems down the road if there are significant changes to the structure of the database required; the manual does cover this, but no formal training was provided.

Most of the consultant's time was spent on system development to provide basic functionality such easy inputting of data and easy report generation for common benefits such as Child Benefit Allowances and Old Age Pensions. As a result, other features, such as managerial reports were not created. This was done to insure data was accurately input into the system in the first place and that this data would accurately be reflected in the voucher reports in order to keep the data in the system up to date and accurate for future managerial reports.

Outer island data from vouchers of current beneficiaries was migrated into the system, but Rarotonga data was not because up-to-date information was not available at migration time. The Senior Welfare Officer will have to prepare the data and import it as outlined in the manual. This can be a useful exercise in itself, as at least one beneficiary receiving benefits on multiple islands was found in the process.

Recommendations

The most important step to building capacity and keeping the database sustainable is simply to use the system. Particularly, making sure the data is up to date will ensure that management can use the data at a later time and that the voucher reports can be verified for accuracy, allowing DSW to gain confidence in using the database. It is often difficult for organizations to adapt to something new, so DSW will have to make a conscious attempt to use the system; otherwise, the system will lay idle until somebody is hired for data entry or it no longer fits the voucher process.

As use continues and users grow more comfortable, the Senior Welfare Officer should experiment with editing reports and forms to customize for new needs in the future. The Senior Welfare Officer should also attempt to train other employees of the ministry or division, particularly in administration, in case something should arise and they are unable to tend to this.

While it is tempting to recommend learning SQL, this is not necessary to use or edit the system (but it is helpful).

Thus, DSW should take the following steps:

1. Use the system weekly in conjunction with the regular Excel voucher reports until DSW gains confidence in the system.
2. Migrate data from the Rarotonga vouchers into the system so it can be used for Rarotonga beneficiaries.
3. Management needs to pressure the officers to use the system.
4. OPM will need to assist DSW in management of the system, as the contractor cannot.

Without a conscious effort to use the system, everybody’s time will have been wasted.

III. Additional Recommendations

A. Integrate Other Programs and Divisions into the Server

Other divisions and programs expressed interest in establishing a database. Indeed, MINTAFF purchased the server in hopes that the entire ministry could utilize it for databases of their own. However, the system is made custom for the use of DSW for benefit administration and would not be much use for other programs and divisions.

While there is a chance that internal capacity will be developed enough within a couple years, a consultant will probably be needed to create the database. In any event, the process will be more efficient if those programs have an explicit (preferably written) idea of what information they will need to keep track of and what they need to learn from that information.

B. Automate Information Tracking

As stated previously, DSW keeps track of changes in the data over time through summaries included with the vouchers. While briefly attempting to include this task, the consultant deemed it beyond the scope of what could be accomplished in the short time in the office. This task would require adding some fields to the database and creating new queries/reports. This is beyond the capacity of MINTAFF and outside help would be needed for this, except for the simplest of queries, such as the number of beneficiaries in a specific location, et cetera.

C. Automate Flagging

The system has no flags built in to catch potential misuse. This is mostly due to time spent on system development and debugging as opposed to adding functionality. However, DSW will have to evaluate exactly what to flag and what would indicate a conflict before any automation can be built into the system. Some conflicts would be easy to find by running regular queries, such as selecting all beneficiaries receiving Child Benefit Allowances with a date of birth less than a particular date. As is, a 44 year old man could be signed up to receive an Old Age 70+ Pension, and nothing would indicate a problem without looking for it.

D. Integrate Databases and Track Beneficiaries

While there is a trend toward centralization, this is still some time off. DSW would like to track total time in country of a beneficiary. While this is easy enough by creating a table within the *welfare* database with the BenID, EntryDate, and DepartureDate, and summing the difference between the EntryDate and DepartureDate for entry of each beneficiary, it seems to be a waste of resources. The Ministry of Immigration and Foreign Affairs (MIFA) already tracks this information and has the capacity to track it much more easily than DSW. It makes more sense to integrate the databases or simply refer to MIFA when this information is needed. As is, the database does allow payments to be suspended by setting a benefit type to 'Inactive.'

E. Generate Management Reports

As stated earlier, the system currently provides basic functionality to generate various voucher payments. Management also desires information for forecasting and annual reports and the system, while able to provide this information, does not currently offer an easy way to find this information without knowing SQL. To facilitate this, management needs to state exactly what they need to learn from the database on a routine basis, and the appropriate query can be written. To run "off-the-cuff" queries based on newly required information by management will require training somebody in SQL or hiring outside help.

F. Retrieve Overpayments

Retrieving overpayments of arrears from beneficiaries incorrectly receiving payments due to mistakes of fraud is a time consuming process for DSW. The system currently offers no functionality for this and an outside consultant would likely be needed if wishing to integrate this action in the future. It is the hope, however, that the system itself will reduce overpayments through mistakes.

G. Develop and Implement a Backup Plan

OPM has been developing a nationwide backup plan, but a ministry-wide backup plan is crucial to data integrity and security. The Computer Man can and should assist with system backup as a whole. MySQL Workbench offers data backup functionality to make backing up the data easy.

H. Upgrade the Server

While the server, as is, should be fine for DSW uses, as other divisions and programs use the server, it will need to be upgraded. Luckily, relatively cheap and easy upgrades will add a lot of computing power:

a. Upgrading to a 64-bit operating system (most new operating systems, including Windows 7 or Linux) will allow the system to utilize all of the RAM in the server and significantly increase responsiveness of the server.

b. Reinstalling MySQL as a *Dedicated* set-up will have the server dedicate almost all of its resources to database work. It is currently installed as a *server* set-up, which allocates a moderate amount of resources, but leaves some for other applications (such as a mail server).

c. Upgrading the power source will have to be done if MINTAFF has any desire to add any hardware to the server, such as another hard disk or another fan.

None of these steps would be prohibitively costly or take much time to implement.

Identify the source of the recommendation—explain why you are dealing with this issue, e.g. a problem or opportunity you identified but didn't work on in the consulting partnership, an issue you observed while there that you think they should address, etc.

Why should they implement this recommendation? Provide a persuasive argument for why the recommendation is important. If the argument is that it would support their mission better, quantify how "better" would be measured (e.g. cost savings or via Return on Investment (ROI)). Or if the argument is for more efficient use of resources (people or funds), quantify how "more efficient" would be measured, e.g. time saved, money saved, or ROI. Think in terms of the same type of concrete outcomes that you presented in your Outcomes Analysis.

What steps should they follow to implement each recommendation? Give concrete actions at a level appropriate for the CP to understand and be able to implement.

What resources will help them implement those recommendations? Why are these resources useful? How should they use them to implement the recommendation?

About the Consultant

Tim Garvey will graduate with a Master's of Science in Public Policy and Management from Carnegie Mellon University at the end of 2012. He has previous experience working with Pacific communities and has also completed a database system for a housing organization in Washington, DC.

Appendix A.

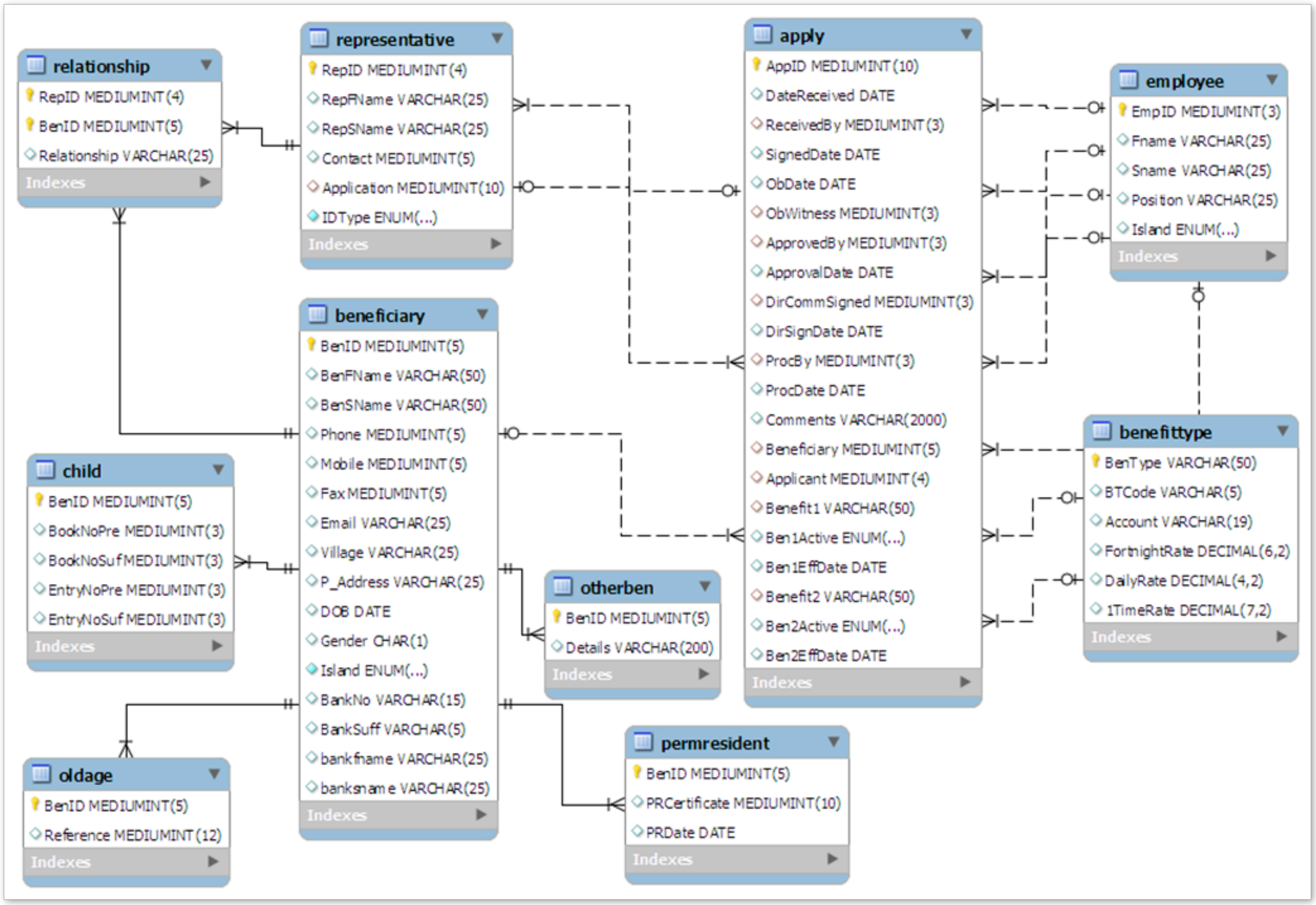
SQL used for creating MySQL Database:

```
DROP Table Employee              CASCADE;
DROP Table Representative        CASCADE;
DROP Table Beneficiary           CASCADE;
DROP Table CountryTime           CASCADE;
DROP Table Relationship          CASCADE;
DROP Table OtherBen              CASCADE;
DROP Table Apply                 CASCADE;
DROP Table BenefitType           CASCADE;
DROP Table Child                 CASCADE;
DROP Table OldAge                CASCADE;
DROP Table PermResident          CASCADE;

CREATE TABLE Employee
(EmpID                   MEDIUMINT(3) AUTO_INCREMENT,
 Fname                   VARCHAR(25),
 Sname                   VARCHAR(25),
 Position                VARCHAR(25),
 Island                  ENUM('Atiu', 'Mauke', 'Mitiaro', 'Takutea', 'Manihiki', 'Nassau', 'Penrhyn', 'Pukapuka', 
'Rakahanga', 'Suwarrow', 'Aitutaki', 'Mangaia', 'Rarotonga', 'Palmerston', 'Manuae') DEFAULT 'Rarotonga',
 CONSTRAINT EmpID_NonNeg_CK CHECK (EmpID > 0),
 CONSTRAINT Employee_EmpID_PK PRIMARY KEY (EmpID));

CREATE TABLE Representative
(RepID                   MEDIUMINT(4) AUTO_INCREMENT,
 RepFName                VARCHAR(25),
 RepSName                VARCHAR(25),
 Contact                 MEDIUMINT(5),
 Application             MEDIUMINT(10),
 IDType                  ENUM('License','Passport','BCertificate','Pending') DEFAULT 'Pending' NOT NULL,
 CONSTRAINT RepID_NonNeg_CK CHECK (RepID > 0),
 CONSTRAINT  Rep_PK PRIMARY KEY (RepID));

CREATE TABLE Beneficiary
(BenID                   MEDIUMINT(5) AUTO_INCREMENT,
 BenFName                VARCHAR(50),
 BenSName                VARCHAR(50),
 Phone                   MEDIUMINT(5),
 Mobile                  MEDIUMINT(5),
 Fax                     MEDIUMINT(5),
 Email                   VARCHAR(25),
 Village                 VARCHAR(25),
 P_Address               VARCHAR(25),
 DOB                     DATE,
 Gender                  CHAR(1),
 Island                  ENUM('Atiu', 'Mauke', 'Mitiaro', 'Takutea', 'Manihiki', 'Nassau', 'Penrhyn', 'Pukapuka', 
'Rakahanga', 'Suwarrow', 'Aitutaki', 'Mangaia', 'Rarotonga', 'Palmerston', 'Manuae') NOT NULL,
 BankFName               VARCHAR(25),
 BankSName               VARCHAR(25),
 BankNo                  VARCHAR(15),
 BankSuff                VARCHAR(5),
 CONSTRAINT BenID_NonNeg_CK CHECK (BenID > 0),
 CONSTRAINT Beneficiary_PK PRIMARY KEY (BenID));

CREATE TABLE CountryTime
(Row                     MEDIUMINT(6) AUTO_INCREMENT,
 Beneficiary             MEDIUMINT(5),
 EntryDate               DATE NOT NULL,
 DepartDate              DATE,
 CONSTRAINT CountryTime_BenID_FK FOREIGN KEY (Beneficiary)
                              REFERENCES Beneficiary (BenID),
 CONSTRAINT Row_NonNeg_CK CHECK (Row > 0),
 CONSTRAINT TIC_DepEnt_CK CHECK (DepartDate > EntryDate),
 CONSTRAINT CountryTime_Row_PK PRIMARY KEY (Row));
```

```
CREATE TABLE Relationship
(RepID                  MEDIUMINT(4),
 BenID                  MEDIUMINT(5),
 Relationship           VARCHAR(25),
 CONSTRAINT  Rel_Rep_FK FOREIGN KEY (RepID)
                              REFERENCES Representative (RepID),
 CONSTRAINT  Rel_Ben_FK FOREIGN KEY (BenID)
                              REFERENCES Beneficiary (BenID),
 CONSTRAINT Rel_Ben_Rep_PK PRIMARY KEY (RepID, BenID));

CREATE TABLE OtherBen
(BenID                  MEDIUMINT(5),
 Details                VARCHAR(200),
 CONSTRAINT  Other_BenID_FK FOREIGN KEY (BenID)
                              REFERENCES Beneficiary (BenID),
 CONSTRAINT Other_Ben_PK PRIMARY KEY (BenID));

CREATE TABLE BenResidence
(BenID                  MEDIUMINT(5),
 ResStatus              ENUM('CI Born CI','CI Born Overseas','PR','Non-CI Born Overseas','Other')
 PassportNo             MEDIUMINT(10),
 PassportExp            DATE,
 BCertificate           VARCHAR(10),
 CONSTRAINT  BO_BenID_FK FOREIGN KEY (BenID)
                              REFERENCES Beneficiary (BenID),
 CONSTRAINT PPNo_NonNeg_CK CHECK (PassportNo > 0),
 CONSTRAINT BO_Ben_PK PRIMARY KEY (BenID));

CREATE TABLE PermResident
(BenID                  MEDIUMINT(5),
 PRCertificate          MEDIUMINT(10),
 PRDate                 DATE,
 CONSTRAINT  PR_BenID_FK FOREIGN KEY (BenID)
                              REFERENCES Beneficiary (BenID),
 CONSTRAINT PRCert_NonNeg_CK CHECK (PRCertificate > 0),
 CONSTRAINT PR_Ben_PK PRIMARY KEY (BenID));

CREATE TABLE OldAge
(BenID                  MEDIUMINT(5),
 Reference              MEDIUMINT(12) ZEROFILL,
 CONSTRAINT OldAge_BenID_FK FOREIGN KEY (BenID)
                              REFERENCES Beneficiary (BenID),
 CONSTRAINT OldAge_BenID_PK PRIMARY KEY (BenID));

CREATE TABLE Child
(BenID                  MEDIUMINT(5),
 BookNoPre              MEDIUMINT(3),
 BookNoSuf              MEDIUMINT(3),
 EntryNoPre             MEDIUMINT(3),
 EntryNoSuf             MEDIUMINT(3),
 CONSTRAINT Child_BenID_FK FOREIGN KEY (BenID)
                              REFERENCES Beneficiary (BenID),
 CONSTRAINT Child_BenID_PK PRIMARY KEY (BenID));

CREATE TABLE BenefitType
(BenType                VARCHAR(50),
 BTCode                 VARCHAR(5),
 Account                VARCHAR(19),
 FortnightRate          DECIMAL(6,2) DEFAULT 0,
 DailyRate              DECIMAL(4,2) DEFAULT 0,
 1TimeRate              DECIMAL(7,2) DEFAULT 0,
 CONSTRAINT BenType_BenType_PK PRIMARY KEY (BenType));

CREATE TABLE Apply
(AppID                  MEDIUMINT(10) AUTO_INCREMENT,
 DateReceived           DATE,
 ReceivedBy             MEDIUMINT(3),
 SignedDate             DATE,
 ObDate                 DATE,
 ObWitness               MEDIUMINT(3),
 ApprovedBy              MEDIUMINT(3),
 ApprovalDate            DATE,
 DirCommSigned           MEDIUMINT(3),
 DirSignDate             DATE,
 ProcBy                  MEDIUMINT(3),
 ProcDate                DATE,
 Comments                VARCHAR(2000),
 Beneficiary             MEDIUMINT(5),
 Applicant               MEDIUMINT(4),
 Benefit1                VARCHAR(50),
 Ben1Active              ENUM('Active', 'Inactive', 'Frozen') DEFAULT 'Active',
 Ben1EffDate             DATE,
 Benefit2                VARCHAR(50),
 Ben2Active              ENUM('Active', 'Inactive', 'Frozen') DEFAULT 'Inactive',
 Ben2EffDate             DATE,
 CONSTRAINT Emp_Received_FK FOREIGN KEY (ReceivedBy)
                              REFERENCES Employee (EmpID),
 CONSTRAINT Emp_Approved_FK FOREIGN KEY (ApprovedBy)
                              REFERENCES Employee (EmpID),
 CONSTRAINT Emp_Witness_FK FOREIGN KEY (ObWitness)
                              REFERENCES Employee (EmpID),
 CONSTRAINT Emp_DirSign_FK FOREIGN KEY (DirCommSigned)
                              REFERENCES Employee (EmpID),
 CONSTRAINT Emp_Proc_FK FOREIGN KEY (ProcBy)
                              REFERENCES Employee (EmpID),
 CONSTRAINT Rep_Apply_FK FOREIGN KEY (Applicant)
                              REFERENCES Representative (RepID),
 CONSTRAINT Ben_Apply_FK FOREIGN KEY (Beneficiary)
                              REFERENCES Beneficiary (BenID),
 CONSTRAINT BenT1_AppBen1_FK FOREIGN KEY (Benefit1)
                              REFERENCES BenefitType (BenType),
 CONSTRAINT BenT2_AppBen2_FK FOREIGN KEY (Benefit2)
                              REFERENCES BenefitType (BenType),
 CONSTRAINT Apply_AppID_PK PRIMARY KEY (AppID));

ALTER TABLE Representative
  ADD CONSTRAINT Apply_Rep_FK FOREIGN KEY (Application)
                                 REFERENCES Apply (AppID);

COMMIT;
```

Appendix B.

Entity-Relationship diagram of database (views not included) from MySQL Workbench:



Appendix C.

SQL for creating views:

Additional Voucher

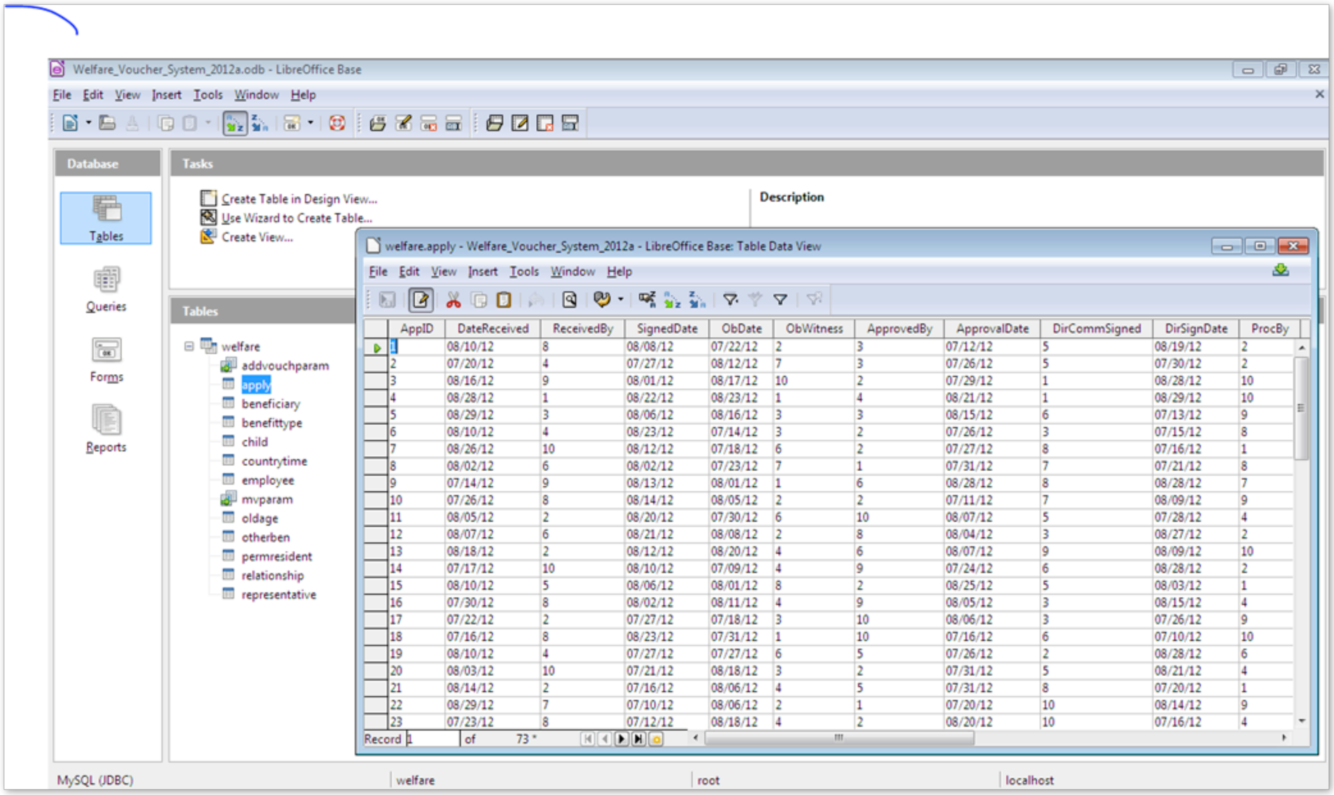
```
CREATE VIEW AddVouchParam AS SELECT BenID ID, BenType Benefit, BTCode Code, Account BCI, CONCAT_WS(',',
BankSName, BankFName) Person, DOB Born, Gender Sex, DirSignDate DirDate, Ben1EffDate EffDate, Island Place,
BankNo Bank, BankSuff Suff, DailyRate Rate, 1TimeRate 1Rate
FROM Beneficiary b, Apply a, BenefitType bt
WHERE b.BankNo IS NOT NULL
AND b.benid = a.beneficiary
AND a.benefit1 = bt.bentype
AND Ben1Active = 'Active'
AND Ben1EffDate IS NOT NULL
UNION
SELECT BenID, BenType, BTCode, Account, CONCAT_WS(',', BankSName, BankFName), DOB, Gender,
DirSignDate, Ben2EffDate, Island, BankNo, BankSuff, DailyRate, 1TimeRate
FROM Beneficiary b, Apply a, BenefitType bt
WHERE b.BankNo IS NOT NULL
AND b.benid = a.beneficiary
AND a.benefit2 = bt.bentype
AND Ben2Active = 'Active'
AND Ben2EffDate IS NOT NULL;
```

Main Voucher

```
CREATE VIEW MVParam AS SELECT BenID Ben, BenType BType, BTCode BenCode, Account BCI, CONCAT_WS(',', BankSName,
   BankFName) Person, DOB Born, Gender Sex, Island Place, FortnightRate Amount, BankNo Bank, BankSuff Suff,
   DirsignDate DirDate, Ben1EffDate EffDate
FROM Beneficiary b, Apply a, BenefitType bt
WHERE b.BankNo IS NOT NULL
AND b.benid = a.beneficiary
AND a.benefit1 = bt.bentype
AND Ben1Active = 'Active'
AND Ben1EffDate IS NOT NULL
UNION
SELECT BenID, BenType, BTCode, Account, CONCAT_WS(',', BankSName, BankFName), DOB, Gender, Island,
   FortnightRate, BankNo, BankSuff, DirSignDate, Ben2EffDate
FROM Beneficiary b, Apply a, BenefitType bt
WHERE b.BankNo IS NOT NULL
AND b.benid = a.beneficiary
AND a.benefit2 = bt.bentype
AND Ben2Active = 'Active'
AND Ben2EffDate IS NOT NULL;
```

Appendix D.

Tables in Base



Welfare_Voucher_System_2012a.odb - LibreOffice Base
File Edit View Insert Tools Window Help
Database
Tables
Queries
Forms
Reports
Tasks
Create Table in Design View...
Use Wizard to Create Table...
Create View...
Description
Tables

- welfare
  - addvouchparam
  - apply
  - beneficiary
  - benefittype
  - child
  - countrytime
  - employee
  - mvparam
  - oldage
  - otherben
  - permresident
  - relationship
  - representative

welfare.apply - Welfare_Voucher_System_2012a - LibreOffice Base: Table Data View

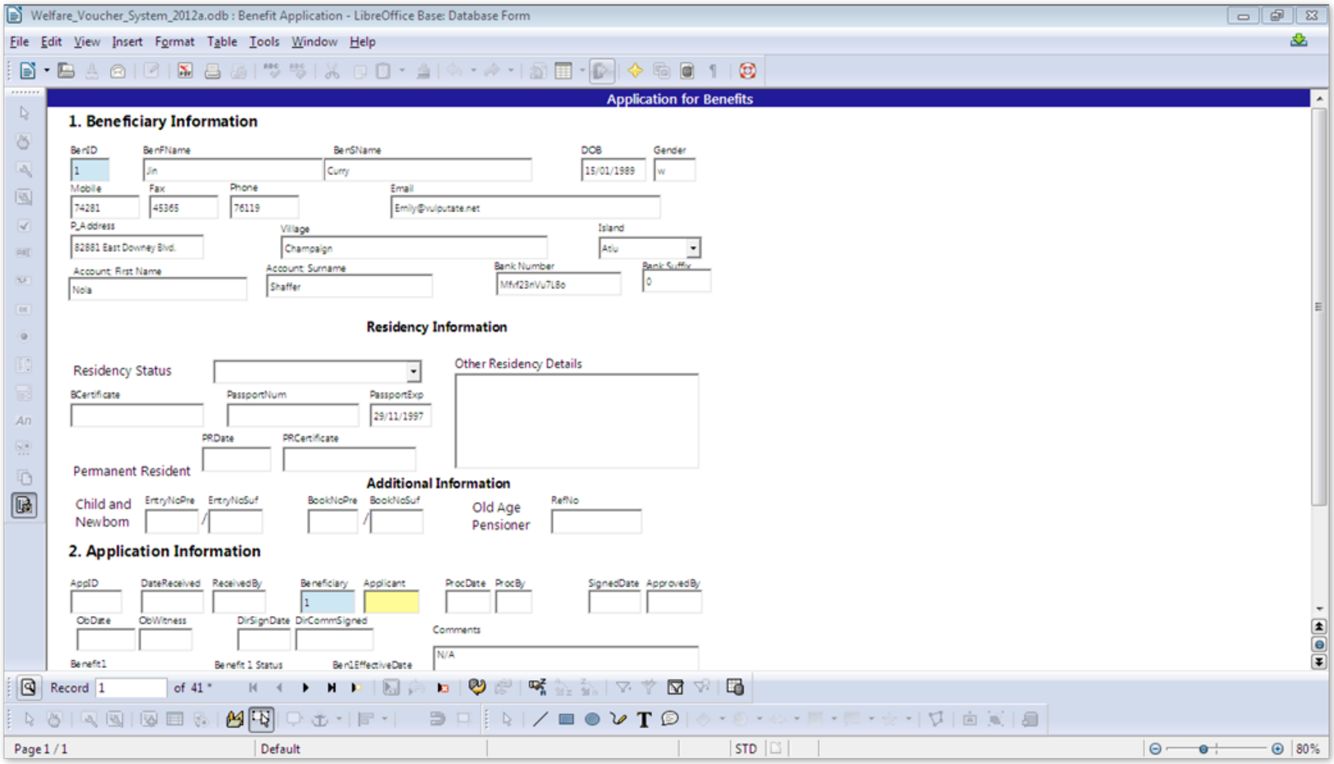
File Edit View Insert Tools Window Help

AppID	DateReceived	ReceivedBy	SignedDate	ObDate	ObWitness	ApprovedBy	ApprovalDate	DirCommSigned	DirSignDate	ProcBy
1	08/10/12	8	08/08/12	07/22/12	2	3	07/12/12	5	08/19/12	2
2	07/20/12	4	07/27/12	08/12/12	7	3	07/26/12	5	07/30/12	2
3	08/16/12	9	08/01/12	08/17/12	10	2	07/29/12	1	08/28/12	10
4	08/28/12	1	08/22/12	08/23/12	1	4	08/21/12	1	08/29/12	10
5	08/29/12	3	08/06/12	08/16/12	3	3	08/15/12	6	07/13/12	9
6	08/10/12	4	08/23/12	07/14/12	3	2	07/26/12	3	07/15/12	8
7	08/26/12	10	08/12/12	07/18/12	6	2	07/27/12	8	07/16/12	1
8	08/02/12	6	08/02/12	07/23/12	7	1	07/31/12	7	07/21/12	8
9	07/14/12	9	08/13/12	08/01/12	1	6	08/28/12	8	08/28/12	7
10	07/26/12	8	08/14/12	08/05/12	2	2	07/11/12	7	08/09/12	9
11	08/05/12	2	08/20/12	07/30/12	6	10	08/07/12	5	07/28/12	4
12	08/07/12	6	08/21/12	08/08/12	2	8	08/04/12	3	08/27/12	2
13	08/18/12	2	08/12/12	08/20/12	4	6	08/07/12	9	08/09/12	10
14	07/17/12	10	08/10/12	07/09/12	4	9	07/24/12	6	08/28/12	2
15	08/10/12	5	08/06/12	08/01/12	8	2	08/25/12	5	08/03/12	1
16	07/30/12	8	08/02/12	08/11/12	4	9	08/05/12	3	08/15/12	4
17	07/22/12	2	07/27/12	07/18/12	3	10	08/06/12	3	07/26/12	9
18	07/16/12	8	08/23/12	07/31/12	1	10	07/16/12	6	07/10/12	10
19	08/10/12	4	07/27/12	07/27/12	6	5	07/26/12	2	08/28/12	6
20	08/03/12	10	07/21/12	08/18/12	3	2	07/31/12	5	08/21/12	4
21	08/14/12	2	07/16/12	08/06/12	4	5	07/31/12	8	07/20/12	1
22	08/29/12	7	07/10/12	08/06/12	2	1	07/20/12	10	08/14/12	9
23	07/23/12	8	07/12/12	08/18/12	4	2	08/20/12	10	07/16/12	4

Record 1 of 73 *

MySQL (JDBC) | welfare | root | localhost

Form in Base



Welfare_Voucher_System_2012a.odb : Benefit Application - LibreOffice Base: Database Form

Application for Benefits

1. Beneficiary Information

BenID	BenFName	BenSName	DOB	Gender
1	Jin	Curry	15/01/1989	w
Mobile	Fax	Phone	Email	
74281	45365	76119	Emily@vulputate.net	
P_Address	Village	Island		
82881 East Downey Blvd.	Champaign	Atiu		
Account: First Name	Account: Surname	Bank Number	Bank Suffix	
Nola	Shaffer	Mfvf23nVu7L8o	0	

Residency Information

Residency Status

Other Residency Details

BCertificate	PassportNum	PassportExp
		29/11/1997

Permanent Resident PRDate PRCertificate

Additional Information

Child and Newborn EntryNoPre / EntryNoSuf BookNoPre / BookNoSuf

Old Age Pensioner RefNo

2. Application Information

AppID	DateReceived	ReceivedBy	Beneficiary	Applicant	ProcDate	ProcBy	SignedDate	ApprovedBy
			1					

ObDate ObWitness DirSignDate DirCommSigned

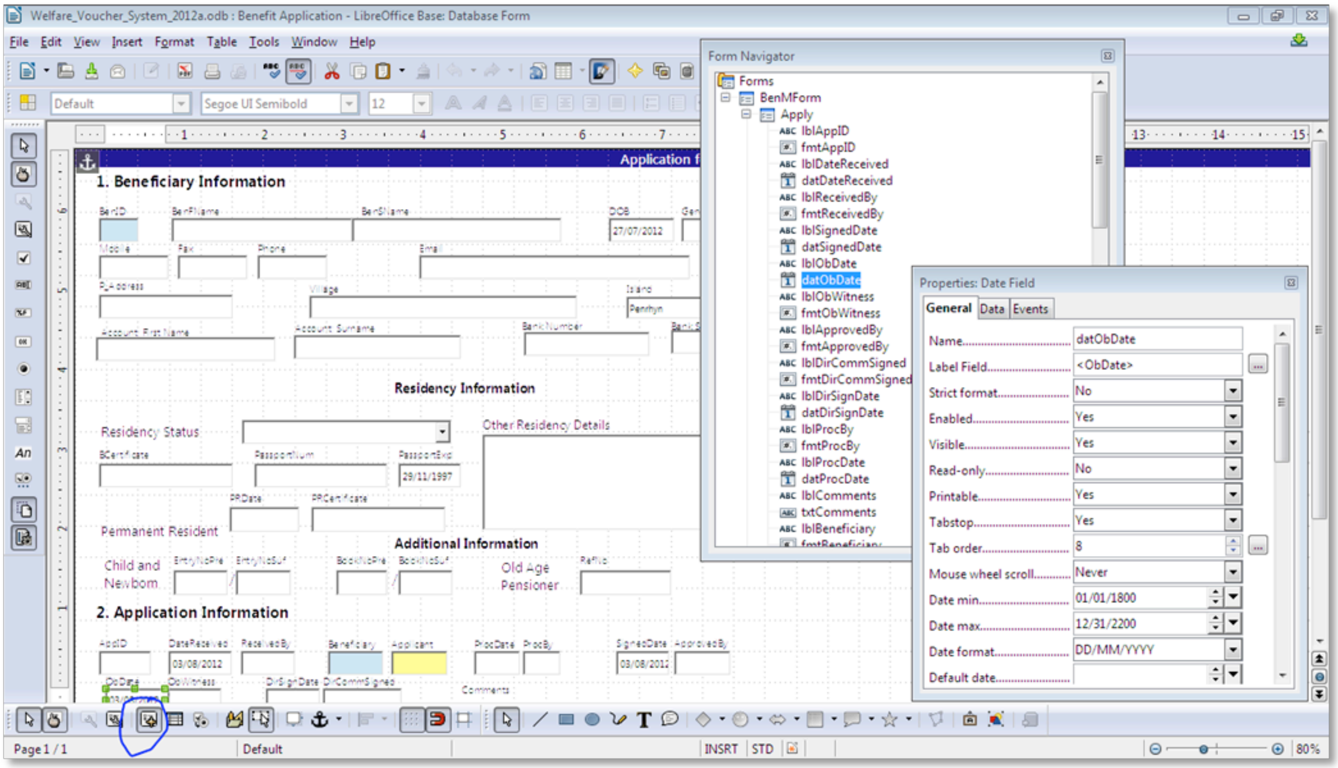
Comments: N/A

Benefit1 Benefit 1 Status Ben1EffectiveDate

Record 1 of 41 *

Page 1 / 1 Default STD 80%

Editing a Form in Base



Welfare_Voucher_System_2012a.odb : Benefit Application - LibreOffice Base: Database Form

Application f

1. Beneficiary Information

BenID BenFName BenSName DOB 27/07/2012 Gen

Mobile Fax Phone Email

P_Address Village Island Penrhyn

Account: First Name Account: Surname Bank Number Bank S

Residency Information

Residency Status Other Residency Details

BCertificate PassportNum PassportExp 29/11/1997

Permanent Resident PRDate PRCertificate

Additional Information

Child and Newborn EntryNoPre EntryNoSuf BookNoPre BookNoSuf Old Age Pensioner RefNo

2. Application Information

AppID DateReceived 03/08/2012 ReceivedBy Beneficiary Applicant ProcDate ProcBy SignedDate 03/08/2012 ApprovedBy

ObDate ObWitness DirSignDate DirCommSigned Comments

Form Navigator

- Forms
  - BenMForm
    - Apply
      - lblAppID
      - fmtAppID
      - lblDateReceived
      - datDateReceived
      - lblReceivedBy
      - fmtReceivedBy
      - lblSignedDate
      - datSignedDate
      - lblObDate
      - datObDate
      - lblObWitness
      - fmtObWitness
      - lblApprovedBy
      - fmtApprovedBy
      - lblDirCommSigned
      - fmtDirCommSigned
      - lblDirSignDate
      - datDirSignDate
      - lblProcBy
      - fmtProcBy
      - lblProcDate
      - datProcDate
      - lblComments
      - txtComments
      - lblBeneficiary
      - fmtBeneficiary

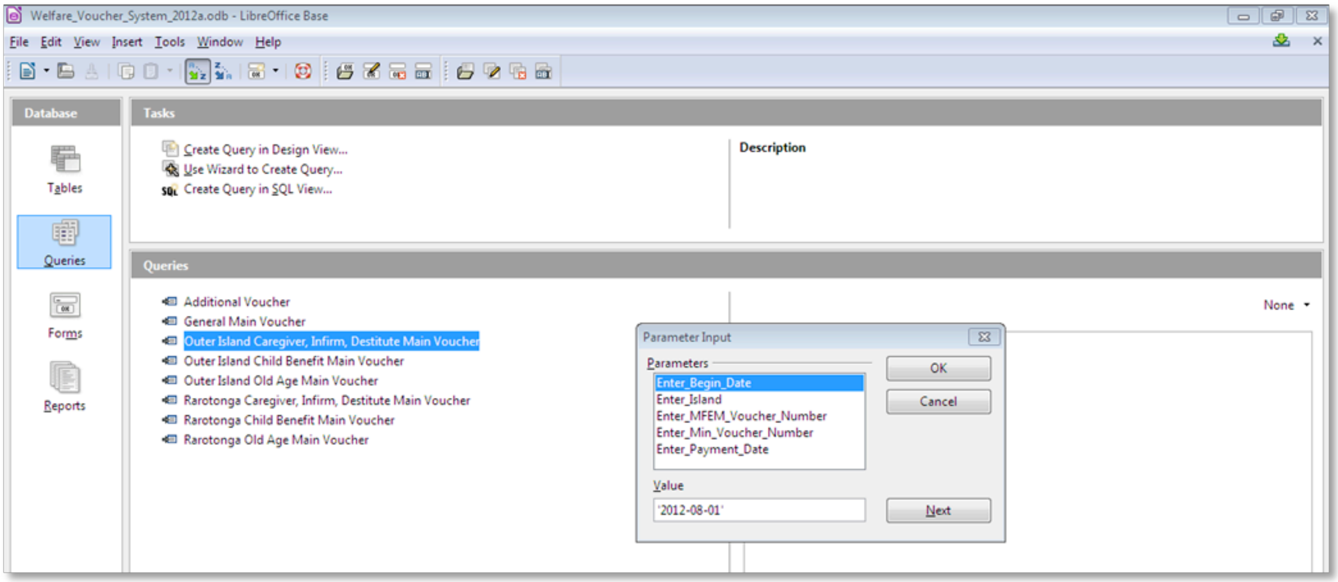
Properties: Date Field

General | Data | Events

Name	datObDate
Label Field	<ObDate>
Strict format	No
Enabled	Yes
Visible	Yes
Read-only	No
Printable	Yes
Tabstop	Yes
Tab order	8
Mouse wheel scroll	Never
Date min	01/01/1800
Date max	12/31/2200
Date format	DD/MM/YYYY
Default date	

Page 1 / 1 Default INSRT STD 80%

Inputting Parameters into a Query in Base


Welfare_Voucher_System_2012a.odb - LibreOffice Base
File Edit View Insert Tools Window Help
Database
Tables
Queries
Forms
Reports
Tasks
Queries
None

Create Query in Design View...
Use Wizard to Create Query...
Create Query in SQL View...

Description

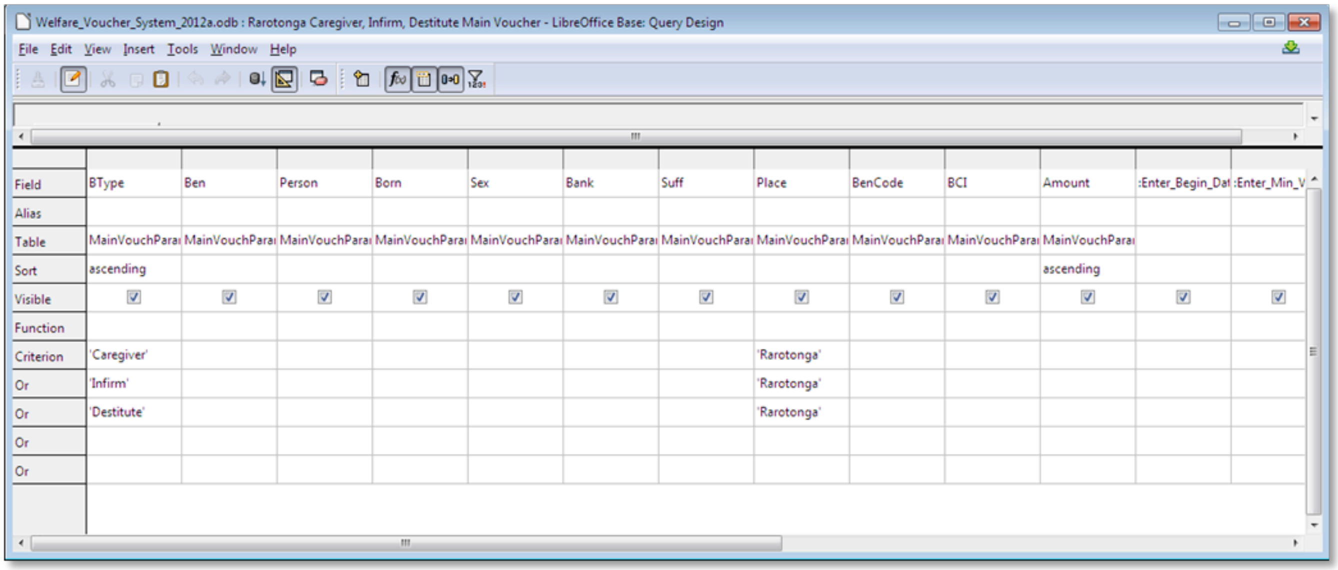
- Additional Voucher
- General Main Voucher
- Outer Island Caregiver, Infirm, Destitute Main Voucher
- Outer Island Child Benefit Main Voucher
- Outer Island Old Age Main Voucher
- Rarotonga Caregiver, Infirm, Destitute Main Voucher
- Rarotonga Child Benefit Main Voucher
- Rarotonga Old Age Main Voucher

Parameter Input

Parameters
Enter_Begin_Date
Enter_Island
Enter_MFEM_Voucher_Number
Enter_Min_Voucher_Number
Enter_Payment_Date

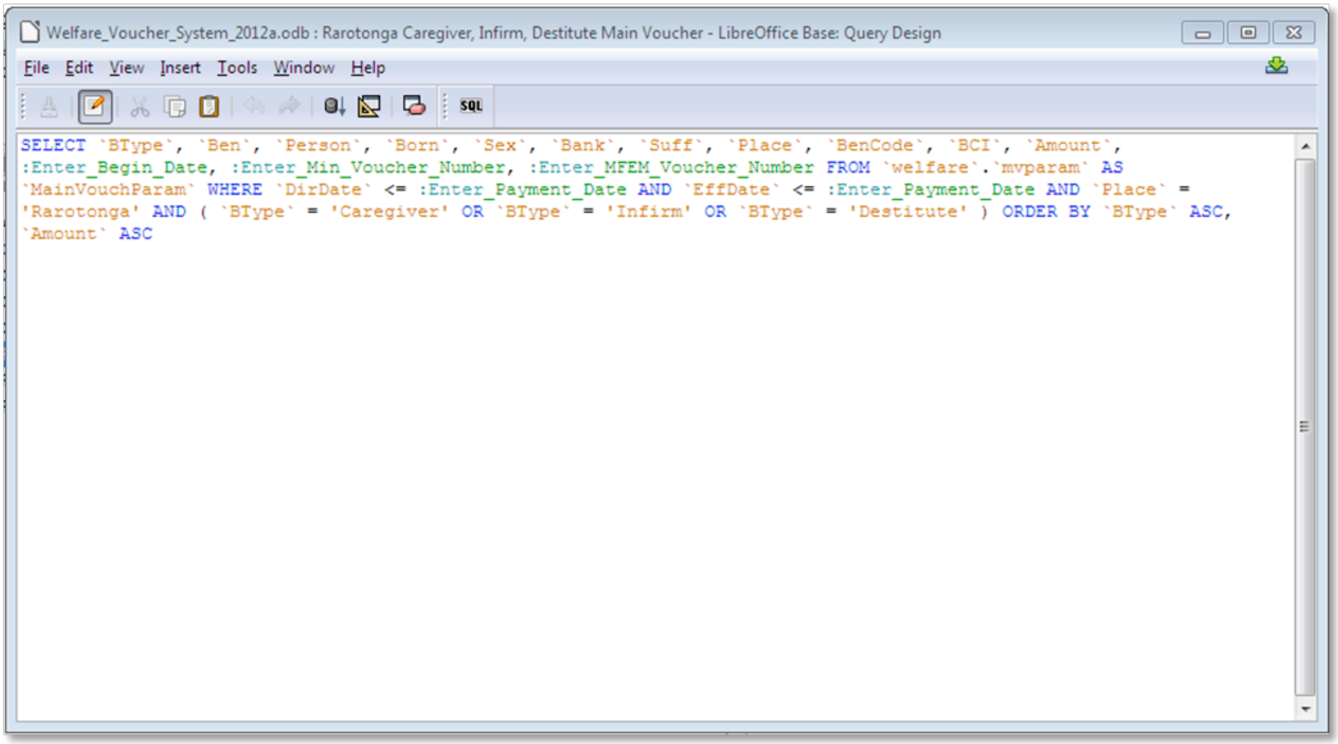
Value
'2012-08-01'

Editing a Query in Design View


Welfare_Voucher_System_2012a.odb : Rarotonga Caregiver, Infirm, Destitute Main Voucher - LibreOffice Base: Query Design
File Edit View Insert Tools Window Help

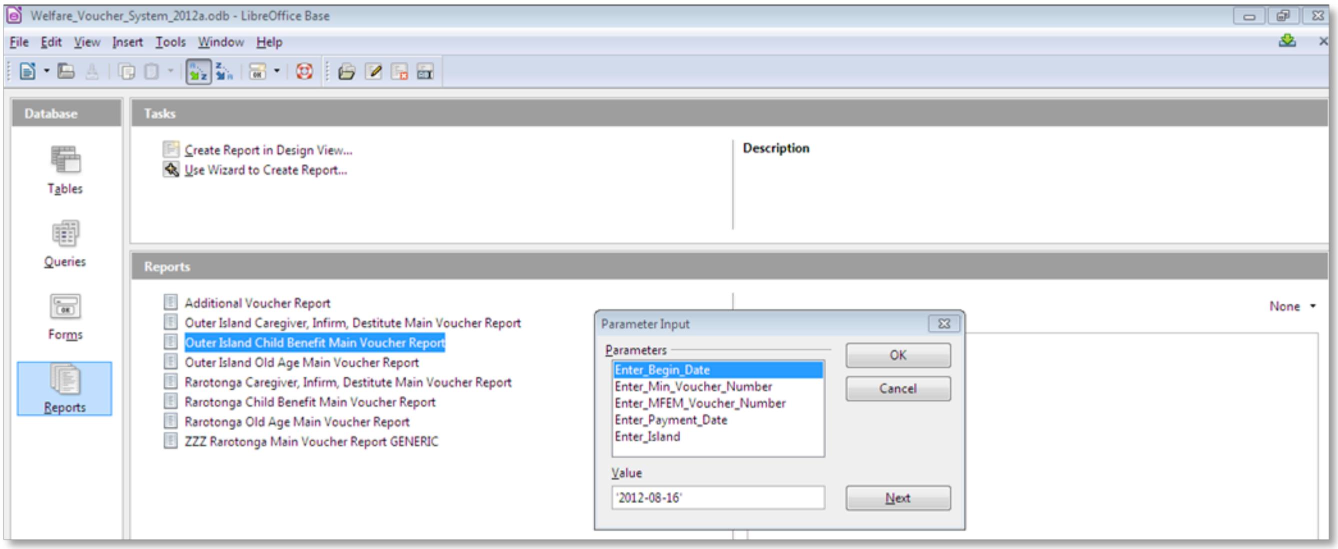
Field	BType	Ben	Person	Born	Sex	Bank	Suff	Place	BenCode	BCI	Amount	:Enter_Begin_Dat	:Enter_Min_V
Alias													
Table	MainVouchPara	MainVouchPara	MainVouchPara	MainVouchPara	MainVouchPara	MainVouchPara	MainVouchPara	MainVouchPara	MainVouchPara	MainVouchPara	MainVouchPara		
Sort	ascending										ascending		
Visible	☑	☑	☑	☑	☑	☑	☑	☑	☑	☑	☑	☑	☑
Function													
Criterion	'Caregiver'							'Rarotonga'					
Or	'Infirm'							'Rarotonga'					
Or	'Destitute'							'Rarotonga'					
Or													
Or													

Editing a Query in SQL View


Welfare_Voucher_System_2012a.odb : Rarotonga Caregiver, Infirm, Destitute Main Voucher - LibreOffice Base: Query Design
File Edit View Insert Tools Window Help
SQL

```
SELECT `BType`, `Ben`, `Person`, `Born`, `Sex`, `Bank`, `Suff`, `Place`, `BenCode`, `BCI`, `Amount`,
:Enter_Begin_Date, :Enter_Min_Voucher_Number, :Enter_MFEM_Voucher_Number FROM `welfare`.`mvparam` AS
`MainVouchParam` WHERE `DirDate` <= :Enter_Payment_Date AND `EffDate` <= :Enter_Payment_Date AND `Place` =
'Rarotonga' AND ( `BType` = 'Caregiver' OR `BType` = 'Infirm' OR `BType` = 'Destitute' ) ORDER BY `BType` ASC,
`Amount` ASC
```

Inputting Parameters for a Report

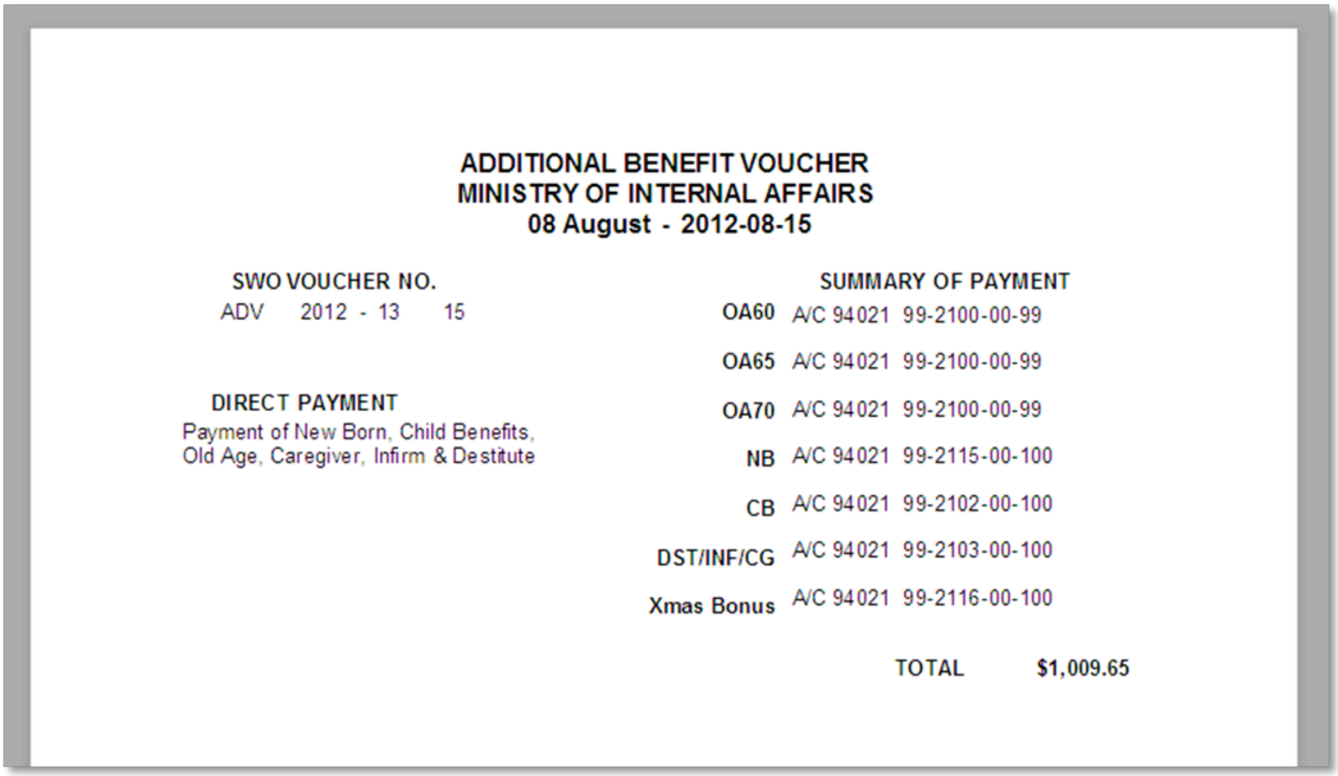

Welfare_Voucher_System_2012a.odb - LibreOffice Base
File Edit View Insert Tools Window Help
Database
Tables
Queries
Forms
Reports
Tasks
Create Report in Design View...
Use Wizard to Create Report...
Description
Reports
None
Parameter Input
Parameters
OK
Cancel
Value
Next

Additional Voucher Report
Outer Island Caregiver, Infirm, Destitute Main Voucher Report
Outer Island Child Benefit Main Voucher Report
Outer Island Old Age Main Voucher Report
Rarotonga Caregiver, Infirm, Destitute Main Voucher Report
Rarotonga Child Benefit Main Voucher Report
Rarotonga Old Age Main Voucher Report
ZZZ Rarotonga Main Voucher Report GENERIC

Parameters:
Enter_Begin_Date
Enter_Min_Voucher_Number
Enter_MFEM_Voucher_Number
Enter_Payment_Date
Enter_Island

Value: '2012-08-16'

A Sample Report



ADDITIONAL BENEFIT VOUCHER
MINISTRY OF INTERNAL AFFAIRS
08 August - 2012-08-15

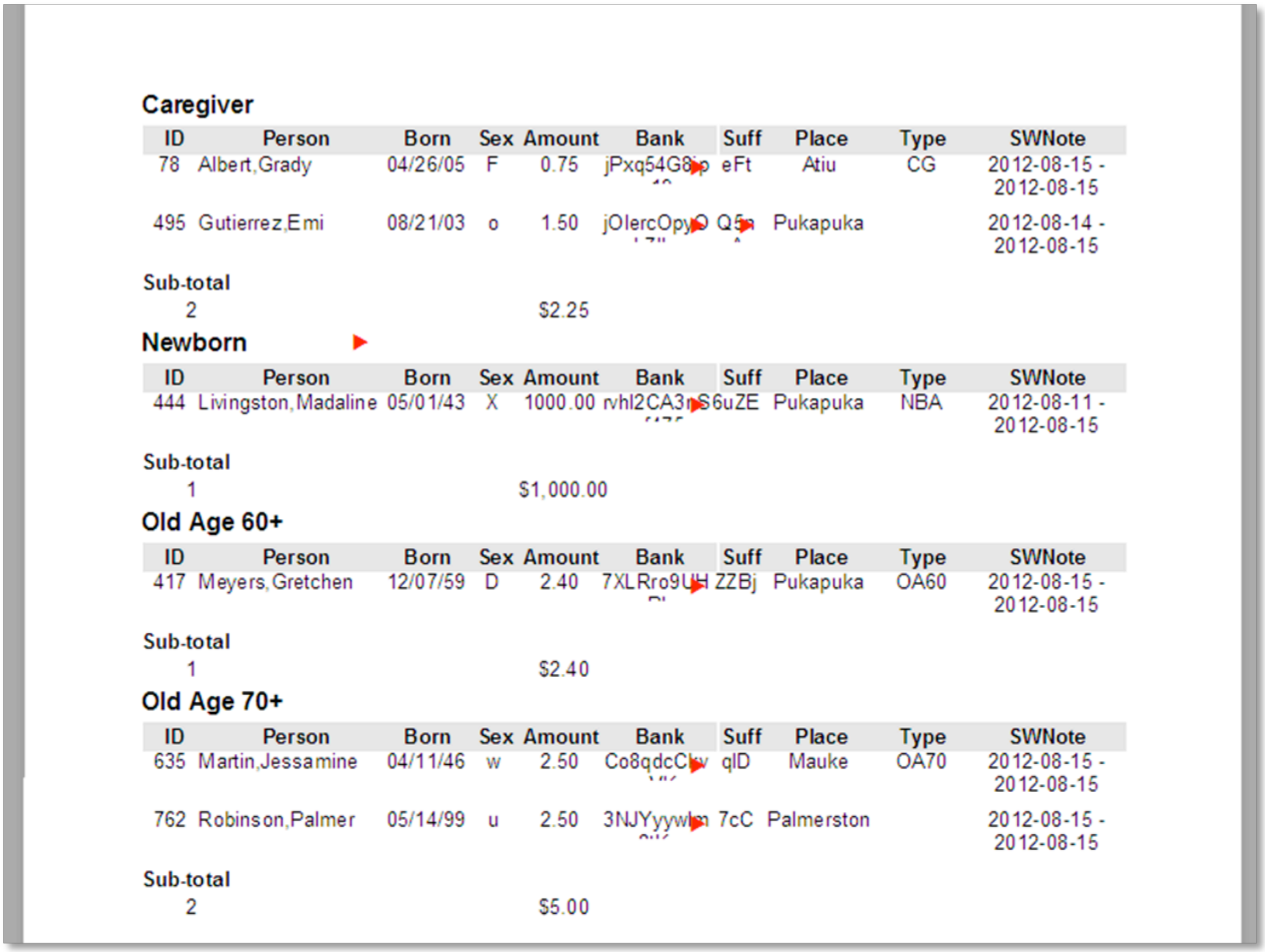
SWO VOUCHER NO.
ADV 2012 - 13 15

DIRECT PAYMENT
Payment of New Born, Child Benefits,
Old Age, Caregiver, Infirm & Destitute

SUMMARY OF PAYMENT

OA60	A/C 94021 99-2100-00-99
OA65	A/C 94021 99-2100-00-99
OA70	A/C 94021 99-2100-00-99
NB	A/C 94021 99-2115-00-100
CB	A/C 94021 99-2102-00-100
DST/INF/CG	A/C 94021 99-2103-00-100
Xmas Bonus	A/C 94021 99-2116-00-100

TOTAL **$1,009.65**



Caregiver

ID	Person	Born	Sex	Amount	Bank	Suff	Place	Type	SWNote
78	Albert,Grady	04/26/05	F	0.75	jPxq54G8jp	eFt	Atiu	CG	2012-08-15 - 2012-08-15
495	Gutierrez,Emi	08/21/03	o	1.50	jOlercOpyO	Q5n	Pukapuka		2012-08-14 - 2012-08-15

Sub-total

2 $2.25

Newborn

ID	Person	Born	Sex	Amount	Bank	Suff	Place	Type	SWNote
444	Livingston,Madaline	05/01/43	X	1000.00	rvhl2CA3n	S6uZE	Pukapuka	NBA	2012-08-11 - 2012-08-15

Sub-total

1 $1,000.00

Old Age 60+

ID	Person	Born	Sex	Amount	Bank	Suff	Place	Type	SWNote
417	Meyers,Gretchen	12/07/59	D	2.40	7XLRro9UH	ZZBj	Pukapuka	OA60	2012-08-15 - 2012-08-15

Sub-total

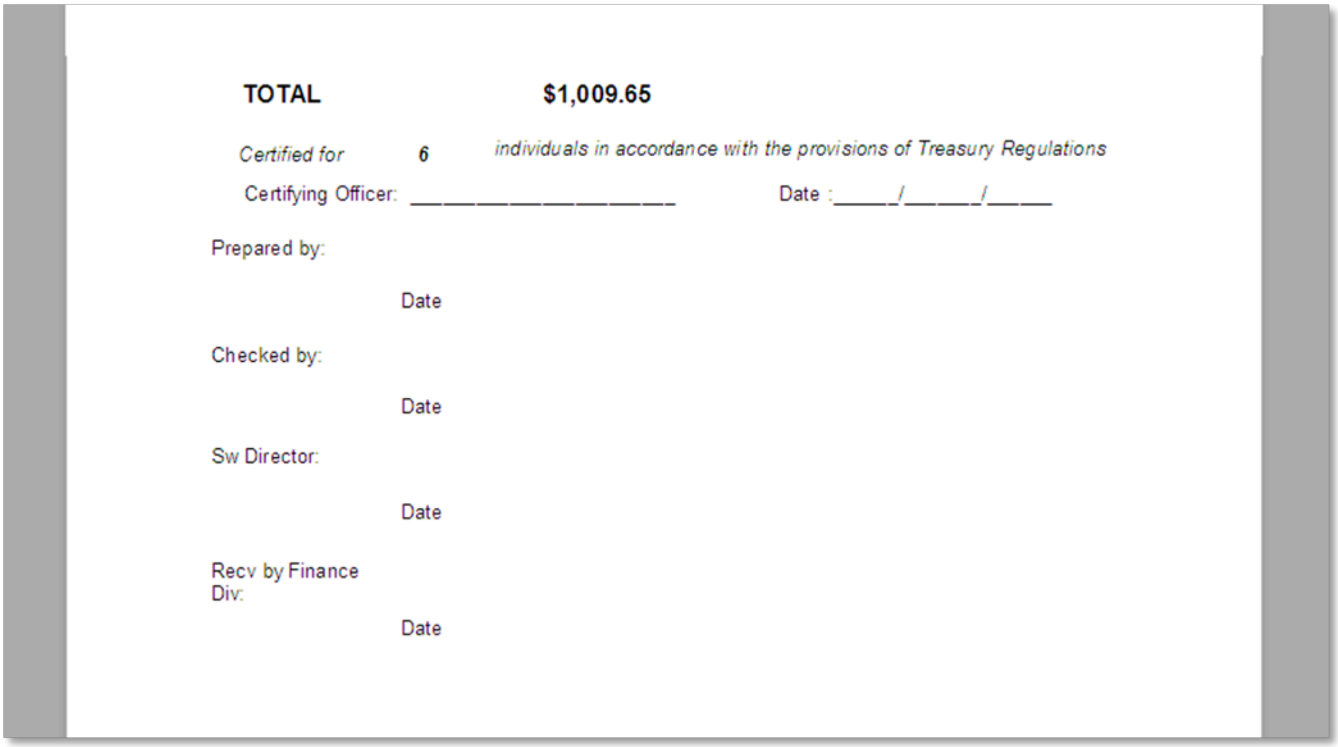
1 $2.40

Old Age 70+

ID	Person	Born	Sex	Amount	Bank	Suff	Place	Type	SWNote
635	Martin,Jessamine	04/11/46	w	2.50	Co8qdcCkw	qID	Mauke	OA70	2012-08-15 - 2012-08-15
762	Robinson,Palmer	05/14/99	u	2.50	3NJYyywlm	7cC	Palmerston		2012-08-15 - 2012-08-15

Sub-total

2 $5.00



TOTAL **$1,009.65**

Certified for ***6*** *individuals in accordance with the provisions of Treasury Regulations*

Certifying Officer: ____________________ Date :_____/_____/_____

Prepared by:

Date

Checked by:

Date

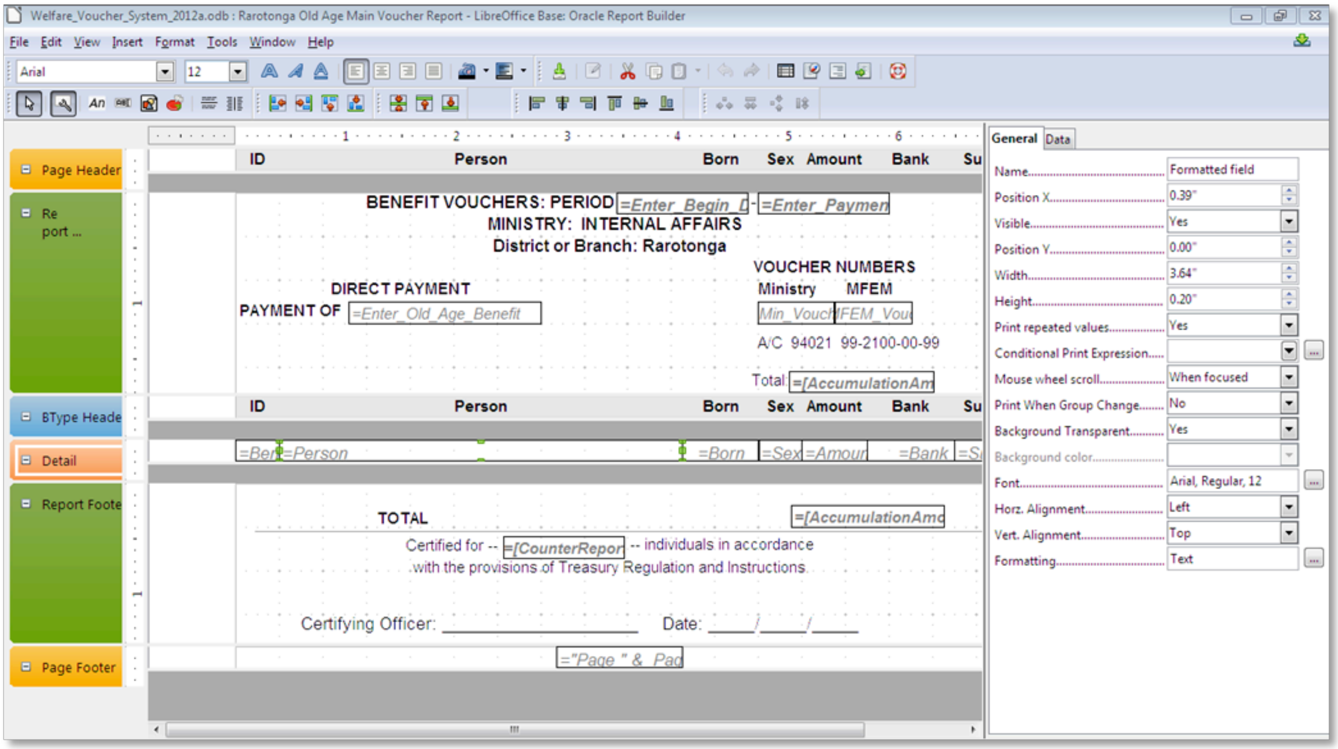
Sw Director:

Date

Recv by Finance Div:

Date

Editing a Report



Welfare_Voucher_System_2012a.odb : Rarotonga Old Age Main Voucher Report - LibreOffice Base: Oracle Report Builder
File Edit View Insert Format Tools Window Help
Arial
12
Page Header
Re port ...
BType Heade
Detail
Report Foote
Page Footer

Page Header: ID Person Born Sex Amount Bank Su

Report Header:

BENEFIT VOUCHERS: PERIOD =Enter_Begin_D - =Enter_Paymen

MINISTRY: INTERNAL AFFAIRS

District or Branch: Rarotonga

VOUCHER NUMBERS

DIRECT PAYMENT

Ministry MFEM

PAYMENT OF =Enter_Old_Age_Benefit

Min_Vouch MFEM_Vou

A/C 94021 99-2100-00-99

Total =[AccumulationAm

BType Header: ID Person Born Sex Amount Bank Su

Detail: =Ben =Person =Born =Sex =Amour =Bank =S

Report Footer:

TOTAL =[AccumulationAmo

Certified for -- =[CounterRepor -- individuals in accordance with the provisions of Treasury Regulation and Instructions

Certifying Officer: ____________________ Date: _____/_____/_____

Page Footer: ="Page " & Pag

General | Data

Name	Formatted field
Position X	0.39"
Visible	Yes
Position Y	0.00"
Width	3.64"
Height	0.20"
Print repeated values	Yes
Conditional Print Expression	
Mouse wheel scroll	When focused
Print When Group Change	No
Background Transparent	Yes
Background color	
Font	Arial, Regular, 12
Horz. Alignment	Left
Vert. Alignment	Top
Formatting	Text

Appendix E.

A copy of the text from the Manual

Division of Social Welfare
Benefit Administration Database Manual

By Tim Garvey
August 2012

Setting up MySQL on the Client Machine

The client machines do not need any MySQL software to use the database, but they do need the connectors installed. All files should be one the storage drive of the database server under Installation Files. Prior to running the MySQL installation program, the following components are required:

Microsoft .NET 4.0 Framework: dotNetFx40_Client_setup can be found online at http://www.microsoft.com/en-us/download/details.aspx?id=17851

Microsoft C++ 2010 Redistributable: vcredist_x86.exe can be found online at http://www.microsoft.com/en-us/download/details.aspx?id=5555

To install the connectors, run the MySQL installer: mysql-installer-5.5.25.0.msi. Click install products and agree to the terms of service. Choose a Custom installation then unclick everything except MySQL Connectors; on the right, only the Connector/ODBC 5.1.11 and Connector/J 5.1.20.0 need be checked. Click next for any additional requirements (there should be none) and proceed clicking Next until finished. Reboot the computer after installation of these connectors.

Setting up LibreOffice's Base 3.5 on Client Machines

LibreOffice is a free, open source (the code is available for anyone to manipulate), package similar to Microsoft Office, with Base as a database client-end similar to Microsoft Access. In the MIntAff context, it is used to interact directly with the MySQL database on the server and no data is stored on the client machine. Installation varies slightly for the various operating systems. Future versions of the software may have different needs and these procedures have to alter to accommodate those needs; version 3.5.5 has been installed on the welfare officer's machines and is considered a "stable" version and ready for organizational use. The software can be finicky and troubleshooting may be required; for this reason, I suggest avoiding upgrades until tested and do not upgrade to anything not described as ready for organizational use by LibreOffice. It is good practice to backup data and set a system restore point prior to any installation and to reboot the computer after each installation. A similar version of Base can be found with the OpenOffice software suite and files should open in both.

As of this writing, the required components to run LibreOffice Base 3.5 can be found here:

MySQL JDBC or ODBC Connector (see above):
The connectors need to be installed prior to setting up Base.

Java: jre-6u33-windows-i586-iftw can be found online at http://java.com/en/download/index.jsp
As of version 3.5 of LibreOffice, Java 7 is not officially supported and earlier builds have security vulnerabilities. Base needs the 32-bit version (x86). The Offline version normally installs better. Java should be installed prior to installing Base.

LibreOffice 3.5 software: LibO_3.5.5_Win_x86_install_multi can be found at http://www.libreoffice.org/download/
If the downloaded file has a .man file extension, change this to .msi to install it on Windows.
LibreOffice Instructions: http://www.libreoffice.org/download/instructions/

Required files as of this writing can be found on the Server under Installation Files.

Base on Windows 7 Ultimate

Windows 7 is 64 bit, and thus requires two versions of Java to run Base. It is currently running on Java 7 build 5 (64 bit) for Windows, which provides a stable platform to operate. Base, however, needs Java 6 build 33 (32 bit) to run, so both will need to be installed prior to installing Base.

In order to access all of the necessary features in Base, you must install Base, Writer, and Calc. If you want the equivalent of other programs similar Microsoft Office (powerpoint, publisher, etc.), choose a typical installation. Otherwise, choose custom and unclick those components besides Base, Calc, and Writer (Impress, etc.) to save resources.

After installation of all required components and rebooting, start Base, create and open a new database; if you get an error message or not, the following will set up Base correctly: click on Tools > Options… and then on Java under the LibreOffice pane on the left. Here, choose to run on version 1.6.0_33 and click OK.

Setting up Base for the MySQL Server

Base connects through the MySQL Server with a driver, or connector; this will typically be ODBC or JDBC. I set up the Base file using JDBC and these instructions are for that connection; however, the file should already be working on the clients as is. For more information on installing the connector, see the MySQL installation section. To set-up the connector, click on Tools > Options… and then on Java under the LibreOffice pane on the left, make sure version 6 build 33 is selected; the, click the Class Path… button on the right and then Add Archive… From here, navigate to the connector used to connect Base to the server; for a JDBC connection, we need the J connector, found at C:\Program Files\MySQL\MySQL Connector J\mysql-connector-java-5.1.20-bin.jar and click OK thrice. It is a good idea to close Base and reboot the system at this point to make sure all changes take place.

Connecting Base to the MySQL Server

Actual networking names and ports may vary slightly. Contact the network administrator with any questions regarding naming.

The database provided should be set-up to connect to the server as is. For creating a new database, click File > New… > Database, Connect to an existing database, choose MySQL and click the Next>>button. Click Connect using JDBC (Java Database Connectivity), and Next>>, the Database name for the Division of Social Welfare is welfare and other databases can be made directly on the server; The Server is DBServer-PC and the Port Number is the default 3306. The MySQL JDBC driver class should read com.mysql.jdbc.Driver and clicking Test class should pop up a window that confirms success; if this window does not pop up, there was likely a step skipped in the installation or set-up in the section above. Use a login and password set-up by the administrator on the server; if you cannot connect, contact the administrator or see the Common Problems, Bug, and Reference section of the manual.

A walkthrough can be found online at http://www.techrepublic.com/blog/doityourself-it-guy/diy-connect-libreoffice-base-to-a-mysql-database/1386 .
To change this information in Base in the future click Edit > Database > Properties and adjust as needed.

Entering Information into the Database System

Within the provided Base file, Welfare_Voucher_System_2012.odb, users have access to the database in different ways. The Tables section allows direct entry and deletion of data in the database. In the Forms section form Benefit Application provides an easy interface for entering data into the MySQL database; double-click the form to open it. The new database only allows applicants to apply for up to two benefits at the same time; if an applicant wishes to apply for more than two benefits, a new application will have to be completed within the database.

Information should be entered in approximately the numbered order provided in the form:

1. *Beneficiary Information*
    a. *Residency Information*
    b. *Additional Information*
2. *Application Information*
3. *Applicant Information*

Only relevant information should be entered (ie. Do not enter Permanent Resident information for somebody not applying as a PR). For details on particular entries, please see the data dictionary. Only certain fields and details will be highlighted here:

- *All dates are formatted as DD/MM/YYYY in the form.*
- *BenID, AppID, RepID, and Beneficiary are auto_incremented and will fill in automatically.*
- *Fields such as ReceivedBy and ProcBy link to the employee table and receive the employee ID for the appropriate employee. See table for emp_ID of each.*
- *The yellow Applicant field should match the RepID from Applicant Information and should be entered by hand after the Applicant Information is entered.*
- *Bank Number, Bank Suffix, Account; First Name, Account: Surname, DirSignDate, Benefit1, Benefit2, Benefit1Active, Benefit2Active, Ben1EffDate, and Ben2EffDate are important fields for calculating payments. If these fields are incorrect or left blank, a payment may not occur.*
- *The blue BenID field under Applicant Information should match the BenID from Beneficiary Information and Beneficiary from Application Information and must be entered by hand.*

Entering New Information for a New Beneficiary

Click the New Record button to enter information for a new applicant. If working on an existing entry and you want to begin an entry for a new beneficiary, click on a field somewhere in the first section of the form prior to clicking the New Record button. The form will automatically insert the information into the proper table within the database.

Adding Information to an Existing Application

For an existing beneficiary, click the Find Record button and enter the appropriate information to find that beneficiary or application. Add information to this record as needed.

Entering a New Application for an Existing Beneficiary

Click on Find Record as above to find the beneficiary. Click on a field in the Application Information portion of the field and then click New Record to add a new application for that beneficiary.

Removing an Application or Beneficiary

There is no need to delete any application or Beneficiary information. Simply mark each benefit for a beneficiary's application(s) as inactive and they will not be computed in the voucher payments. If a deletion is absolutely necessary, click on an active field under Beneficiary Information and click Delete Record to delete all information on that beneficiary. This does not remove their applications; applications need to be deleted by clicking an active field in the Application Information prior to the Delete Record button.

Editing Forms

Right-click on Benefit Application and then click Edit to edit the form. This will allow you to adjust spacing, add and delete fields, and edit how particular fields interact with the database.

Adjusting Spacing and Placement

To move a field, click on the arrow icon on the bottom left of the window, click on what you want to move, and drag it to where you want it. Multiple items can be moved by clicking and dragging a box around them, letting go of the drag, and then clicking and dragging those items where you want. Some items are "grouped," meaning that they will move as a group. To ungroup items, right-click on the group, then Group > ungroup. You can regroup by making a box around all the items you want grouped and right-clicking again.

Editing Fields

The name above or to the side of the text box is a label for that textbox. To manipulate either the label or the text box, open the Form Navigator (the fifth icon from the left with a star at the bottom of the window). Click on whatever you wish to edit; if two items in the navigator are highlighted (the text box and its label), click on whichever item you wish to edit in the navigator. Then, right-click on said item and choose Properties. The General tab in Properties allows you to edit things like date and text format, tab order, display, associated label, or label name (the name in the organizational tree of the form navigator is just for organizational purposes within the navigator). The Data tab edits how the field interacts with the MySQL database; the Data field is the column in MySQL that the information will be placed into and the table will be determined by where that field lies within the organizational tree of the form navigator. The Events tab is for more advanced actions and shouldn't be necessary for any MIntAff needs.

**** Special care should be taken when adding fields as to where in the Form Navigator tree that field lies, as this will effect what table this information goes into and is not always intuitive.***

Adding Fields

One way to add a field is to click the icon on the left of what type of field you wish to add and drawing a box on the form where you want it. You will then need to adjust the formatting, where in the navigator tree this field is, and what column the field connects to in MySQL. A simpler way is to copy and paste a field of the same data type and in the same table you wish to input data to (the paste occurs directly on top of the field you copied, so you have to click that field and move the new copy where you wish), and then edit that field as needed. To connect to a new column, that column must be made in the MySQL server before adding it in Base.

Form Bugs

Occasionally, the bottom of the form may become "jumbled." To fix this, adjust the jumbled portion per the Adjusting Spacing and Placement section above.

Loading a lot of Data into a Table

Occasionally, there may be need to input large amounts of data at once, such as from an Excel file. It does not make sense to use the form in this case and input each piece individually. The Tutorial section of the MySQL documentation (under …/MySQL/Documentation/) gives excellent instructions on how to do this.

To summarize, however, you have to create an Excel spreadsheet and paste in the columns in the same order as they are in the database table (click on whichever table you wish to see the order), and paste the information into that table, placing a \N wherever data is not present to represent NULL values. For example, when importing data from vouchers from Rarotonga, you would want to make a spreadsheet for Beneficiaries, consisting of first name, surname, DOB, Gender, Island, BankNo, and BankSuff; the Island would have to be copied into each row, and you would set the contact information (phone numbers and postal address) as \N because that information is not available from a voucher and you would need to set the BankFName and BankSName the same as the BenFName and BenSName. The DOB would also have to be in a format ready for MySQL, namely YYYY-MM-DD; an easy way to do this is to format the DOB column as a custom format and change it to YYYY-MM-DD, but be sure to pay attention, as some applicants may have their DOB input in an incorrect format to begin with. Though IDs are auto_incremented, it is often prudent to include these into the spreadsheet so they can be coordinated with the foreign keys in other tables, such as the beneficiary field in the Apply table has to match the BenID field in the Beneficiary table.

In the case of importing data from a voucher, another spreadsheet would have to be made for the Apply table consisting of all the relevant dates outlined above (DirSignDate, etc… I set the default date of the outer islands to 2012-06-01), the appropriate beneficiary (BenID copied and pasted from the Beneficiary spreadsheet), the benefit type copied down for each beneficiary, and setting it to active, and then sorting that data to find duplicate entries, taking the second entry and setting it as Benefit2 and Active, and then deleting the duplicate. After all the spreadsheet preparation, the files need to be Save As'd a tab delimited document (remove column names). It is a good idea to make a smaller version of 10 or so rows as a test before importing everything.

Once files are ready, they need to be connected to the server (USB, placed on the server, or through the network), and the MySQL command prompt begun. After logging it, the procedure is:

```
mysql> use welfare
Database changed
mysql>  LOAD DATA LOCAL INFILE 'F:/whereverfileis/beneficiaryimport.txt' INTO TABLE beneficiary;
```

Notice the '/' instead of the normal '\' used in Windows. Common errors are that the order of the columns to not match the order in the table or that the foreign keys do not match, all of which need to be solved in the spreadsheet prior to importing. Obviously, this is all a lot of work.

Deleting a lot of data from a Table

Aside from deleting data in forms, as mentioned above, you can open any table in Base and highlight those rows you wish to delete and press the delete key or right click on the highlighted rows and click on delete. You can narrow down these results using filters by clicking on the top-right icon in the table window. This only deletes the rows from the particular table you have open and not tables that are linked to it; you also will not be able to delete rows that are being linked to from another table (for instance, you could not delete a beneficiary that has a row in the Apply table linking to them.

You can also delete directly in MySQL through the command prompt or MySQL Workbench. This is useful when importing test data and seeing that it did not load correctly. The syntax is:

```
mysql>  use welfare
Database changed
mysql>  delete from whatevertable where tableID > whatever_number_you_began_with AND tableID <
whatever_number_you_ended_with;
```

The tableID can be any column in the table, including text, and it can > 4, < 4, = 4, <= 4, >=4, = 'text', = 'YYYY-MM-DD', > 'YYYY-MM-DD', etc.

Reports Overview

Reports are nothing but formatted queries. A query is a request of information from the My SQL database in SQL syntax; this information is provided as another [temporary] table. Standard syntax is:

```
SELECT something(s)
FROM some_table(s)
WHERE some_condition(s)_is_met;
```

The MySQL Tutorial in the documentation contains several common queries and is a better reference for SQL than this manual is attempting. Queries can get very complicated, consisting of queries of queries, views, and multiple tables joined together, but you will not "break" anything in the database from running query, as it does not alter any data within the database.

Welfare Queries

To view the queries in the welfare system, click on the Queries icon on the left pane of Base. Right-click on the desired query; clicking Edit will open the query in Design Mode, which attempts to make structuring the query easier for someone not versed in SQL. Edit in SQL View will present the query in SQL.

When altering or creating a new query off of an old one, it is good practice to make a copy of the query and manipulate the copy, rather than the original. All queries within the welfare system request information from one of two views.

Welfare Views

A view is like a query built into the database. This system has two views: addvouchparam(which is for additional vouchers) and mvparam (which is for main vouchers). Using views helps speed up the system and simplifies the queries, because all the queries request information from one of these views. The syntax for creating these views can be found in the SQL for Views section.

Addvouchparam

The addvouchparam view provides the following from the database: BenID, BenType, BTCode, Account, BenSName, BenFName, DOB, Gender, DirSignDate, Ben1EffDate/Ben2EffDate, Island, BankNo, BankSuff, DailyRate, 1TimeRate. The Additional Voucher query requests information from this view and performs the required calculation based on the data input by the user.

Mvparam

The mvparam view provides the following from the database: BenID, BenType, BTCode, Account, BenSName, BenFName, DOB, Gender, Island, FortnightRate, BankNo, BankSuff, DirsignDate, Ben1EffDate/Ben2EffDate. This differs from addvouchparam in that no calculations are done in the query, the appropriate rate is simply returned.

Running Voucher Reports

Running voucher reports is as simple as clicking Reports in the right pane and double clicking the type of voucher report you want. A window will pop up asking for relevant information for the report and the report will be generated after the information.

The required date format is YYYY-MM-DD (hyphens and 0's included) and cannot be entered another way. The voucher will break down the payments by benefit type and provide subtotals for the number of beneficiaries and total payment for each benefit type as well as provide grand totals.
Red triangles in reports means there is more information than the space allows. You can click on the triangle to see the information, or you can edit the report to make that section larger.
For additional vouchers, both the beginning and reporting date are important for calculating the appropriate benefit amount. The beginning date is not important for the main vouchers and anything can be input, but this will show up on the report.

Due to a bug in Base, generating reports as spreadsheets requires inputting the parameters twice.
All reports are generated as read-only, but can be saved for later manipulation under \\192.168.0.1\data\#internal affairs\Social Welfare Vouchers\Database Vouchers\. Each report will be separate so a strict naming convention will assist with organization.

Generating Additional/Loose Vouchers

Click on the Reports button in the left pane of Base and double click on Additional Voucher Report to open the additional voucher. A window will pop up asking the user to Enter_Voucher_Number, Enter_Report_Date and Enter_Begin_Date. The report date is the date the voucher will be generated and pay the beneficiary up to and including that date; the begin date is the date at which benefit must have been approved after to be put on the Additional Voucher report and they are entitled to payment on that day as well. For example, a beneficiary that was approved on 10-08-12 (a Friday) would likely have an approval cut off (Begin Date) of 09-08-12 (the Thursday before) and would receive daily payments for every day from the approval date (inclusive) and the date the additional voucher is created (likely 15-08-12, the next Wednesday, inclusive), or 6 days: 10, 11, 12, 13, 14, 15. The voucher number is whatever numbered voucher is being generated.

Generating Main Vouchers

Main vouchers work in a similar fashion to that above, where you enter the report dates, except there is no cut-off date for approval. Everyone that has an 'Active' benefit status, is approved, and has an effective date prior to the main voucher report will receive the fortnight rate for the respective benefit. Double click on the appropriate report for location and benefit type to generate the desired report.
Main vouchers also require entering information for the voucher, so a separate voucher will need to be created for each benefit type and island. The island parameter needs to be exact so be sure to type the island in correctly, or a blank report will be generated.

Editing Reports

Like above, right click on the report and then Edit to edit that report. Editing a report only allows you to edit the format of the report and the information that goes into it from the appropriate query. To edit individual reports, you will first have to save them someplace so they are no longer read only.
A bug exists in that attempting to resize a textbox will often cause Base to crash. To make a textbox larger, you may have to create a new one from scratch. This is very annoying.

MySQL Workbench

MySQL Workbench is a tool on the server for database administration, design, and querying.

SQL Development and Editing Tables

Double clicking the instance will pop up a window for running queries and editing tables. To create a table, right click on Tables under the welfare database on the left and click on create table.
Right-clicking on a desired table and then on edit table data will allow direct manipulation of data within that table. To change the table itself, such as adding columns, changing datatypes, names, column order, etc., click on Alter Table… . Then, click on the two stacked arrows located in the top-right of the pane to see table details. Details on the options available are online by searching for something like "MySQL Workbench tutorial."

Data Modeling

This section is for database planning and will be of little use to Welfare, except for visualizing how tables within the system are related to eachother.

Server Administration

Double clicking the server on the right will allow you to manipulate user privileges, import/export data, view system logs, and toggle server options

Users and Privileges

Click this to add or alter a user. Click Add Account to create a new user or click on a user to alter their set up.

Common Problems, Bugs, and Reference

All system components are free, so they may have more issues than something commercial.

Trouble Logging In

The server is set to go to sleep after several hours of non-use. Turning it on will solve this. Also, make sure that Base is attempting to connect to the proper database. Otherwise, each user has a unique login, so making sure the login is accurate will assist with this. If there are still issues, there may be a problem with the number of simultaneous logins, number of logins/queries in an hour, etc. Contact the database administrator (Helina) for issues regarding this.

A Beneficiary Doesn't Appear in a Report

Make sure that the following information is entered and accurate: BankNo (bank account number), BankSuff (bank account suffix), bankfname (bank account first name), banksname (bank account surname), DirSignDate (date director signed approval), Benefit1 (benefit type), ben1active (is set to active if beneficiary is receiving benefits, Ben1EffDate (date benefit becomes) effective, and the matching columns for Benefit2.

Base Keeps Crashing

Base can sometimes crashes. Ordinarily, this will not be common, but if it becomes a regular issue, the user profile may need to be reset.   Instructions on resetting the profile can be found at http://user.services.openoffice.org/en/forum/viewtopic.php?f=74&t=12426 . This will delete any information such as user names, etc. in Base, but should not affect the system otherwise. If this does not help, reinstalling Base may help. If there are still issues, upgrading the computer or finding another front end (which will require setting up all the reports, etc. from scratch) will have to be done.

Base Crashes when Resizing Text Boxes

This is a bug and cannot be avoided. You will have to create a new textbox and delete the old one.

SQL for Views

AddVouchParam

```
CREATE VIEW AddVouchParam AS SELECT BenID ID, BenType Benefit, BTCode Code, Account BCI, CONCAT_WS(',',
BenSName, BenFName) Person, DOB Born, Gender Sex, DirSignDate DirDate, Ben1EffDate EffDate, Island Place, BankNo
Bank, BankSuff Suff, DailyRate Rate, 1TimeRate 1Rate
FROM Beneficiary b, Apply a, BenefitType bt
WHERE b.BankNo IS NOT NULL
AND b.benid = a.beneficiary
AND a.benefit1 = bt.bentype
AND Ben1Active = 'Active'
AND Ben1EffDate IS NOT NULL
UNION
SELECT BenID, BenType, BTCode, Account, CONCAT_WS(',', BenSName, BenFName), DOB, Gender,
DirSignDate, Ben2EffDate, Island, BankNo, BankSuff, DailyRate, 1TimeRate
FROM Beneficiary b, Apply a, BenefitType bt
WHERE b.BankNo IS NOT NULL
AND b.benid = a.beneficiary
AND a.benefit2 = bt.bentype
AND Ben2Active = 'Active'
AND Ben2EffDate IS NOT NULL;
```

MVParam

```
CREATE VIEW MVParam AS SELECT BenID Ben, BenType BType, BTCode BenCode, Account BCI, CONCAT_WS(',', BankSName,
BankFName) Person, DOB Born, Gender Sex, Island Place, FortnightRate Amount, BankNo Bank, BankSuff Suff,
DirsignDate DirDate, Ben1EffDate EffDate
FROM Beneficiary b, Apply a, BenefitType bt
WHERE b.BankNo IS NOT NULL
AND b.benid = a.beneficiary
AND a.benefit1 = bt.bentype
AND Ben1Active = 'Active'
AND Ben1EffDate IS NOT NULL
UNION
SELECT BenID, BenType, BTCode, Account, CONCAT_WS(',', BankSName, BankFName), DOB, Gender, Island,
FortnightRate, BankNo, BankSuff, DirSignDate, Ben2EffDate
FROM Beneficiary b, Apply a, BenefitType bt
WHERE b.BankNo IS NOT NULL
AND b.benid = a.beneficiary
AND a.benefit2 = bt.bentype
AND Ben2Active = 'Active'
AND Ben2EffDate IS NOT NULL;
```

Resources for Help

I have referred to different sources of assistance throughout the manual. Here is a compiled list of resources.

Web Search

The simplest thing most people will do is to search online for assistance, but you need to know what exactly to search for. In Base, a search for something like “LibreOffice Base parameters in queries,” if you wanted to find out more about using parameters in queries, may be useful. It also helps to search for Open Office instead of LibreOffice because there is a larger user base (pun intended), and the two versions are essentially the same. Most results will come in the form of bulletin boards and forums like Stackoverflow, many of which are blocked. Access to these will help in problem solving.

Likewise, for MySQL and SQL issues, search for something like “MySQL creating views” to find out how to create a view.

Web Sites

LibreOffice has a site with different resources at http://www.libreoffice.org/get-help/ .
For advanced problems, you can try posting the problem at Open Office’s web forum: http://user.services.openoffice.org/en/forum/index.php . This forum is for advanced problems, and is answered freely by other users of Base, so the level assistance varies. Paid support is available as well.

Documentation

The MySQL reference manual is found in the MySQL/Documentation directory on the server, and can be installed with the installer. It is nearly 3800 pages and is available in both a4 and letter format. The tutorial is probably the most useful, with the table of contents of more limited use. The manual is very detailed and can be confusing for the novice user. Using CTRL+F to search for particular language in the manual is sometimes helpful but slow.

Base also has a manual called Getting Started with Base. This is just what it sounds like and covers novice issues in a clear manner. Since it is for the novice, the level of help does not get very in depth.

People in Rarotonga

OPM has stated they will "support" the system, so contacting Pua or Mitch may be helpful, if not just to find somebody else on the island to help.

Khamla Narith works in the Justice Division and has done database work for the Division of Child and Family services. He may offer paid support for the MySQL portion of the system.
The Computer Man can help with networking and server issues, but is unable to offer any sort of database support.

People Outside of Rarotonga

As stated earlier, paid support is available online through the forums above. Caution should be taken with this support, as they may have access to sensitive information. The Computer Man can set up the server to offer remote access from overseas, which would make things easier for any support.
Carnegie Mellon University may be able to offer additional support in the future by either assigning a team of undergraduate students or providing a technology consultant in the winter. This assistance would have to be for a specific and sizeable project, however.

The system developer is able to offer limited assistance from overseas, as time allows, for smaller issues. Email the developer with questions.