TravelCCM

0.5.3

Generated by Doxygen 1.7.4

Mon Dec 5 2011 14:58:58

Contents

1 TravelCCM Documentation

1.1 Getting Started

- Main features of TravelCCM

- Installation

- Linking with TravelCCM

- Users Guide

- Tutorials

- Copyright and License

- Make a Difference

- Make a new release

- TravelCCM People

1.2 TravelCCM at SourceForge

- Project page

- Download TravelCCM

- Open a ticket for a bug or feature

- Mailing lists

- Forums

  - Discuss about Development issues
  - Ask for Help
  - Discuss TravelCCM

1.3 TravelCCM Development

- Git Repository (Subversion is deprecated)

- coding_rules

- Documentation Rules

- Test Rules

1.4 External Libraries

- Boost (C++ STL extensions)

- Python

- MySQL client

- SOCI (C++ DB API)

1.5 Support TravelCCM

1.6 About TravelCCM

TravelCCM aims at providing a clean API, and the corresponding C++ implementation, for choosing one item among a set of travel solutions, given demand-related characteristics (e.g., Willingness-To-Pay, preferred airline, preferred cabin, etc.). N

The TravelCCM C++ library implements some simple Customer Choice Models (CCM), as referenced in the literature (PhD dissertations at MIT, for instance: http://dspace.mit.edu).

The TravelCCM C++ library exposes a simple, clean and object-oriented, API. For instance, the choose() method takes, as input, both a structure representing the travel request (e.g., from Washington, DC, US, to Beijing, China, on the 25th of May) and a list of travel solutions (as provided by the Airline Schedule Manager project: http://sourceforge.net/projects/a and yields, as output, the chosen item.

The output can then be used by other systems, for instance to book the corresponding travel or to visualise it on a map and calendar and to share it with others.

TravelCCM makes an extensive use of existing open-source libraries for increased functionality, speed and accuracy. In particular `Boost` (*C++ STL Extensions*) library is used.

The TravelCCM project originates from the department of Operational Research and Innovation at `Amadeus`, Sophia Antipolis, France. TravelCCM is released under the terms of the `GNU Lesser General Public License` (LGPLv2.1) for you to enjoy.

TravelCCM should work on `GNU/Linux`, `Sun Solaris`, Microsoft Windows (with `Cygwin`, `MinGW/MSYS`, or `Microsoft Visual C++ .NET`) and `Mac OS X` operating systems.

Note

(N) - The TravelCCM library is **NOT** intended, in any way, to be used by airlines for production systems. If you want to report issue, bug or feature request, or if you just want to give feedback, have a look on the right-hand side of this page for the preferred reporting methods. In any case, please do not contact Amadeus directly for any matter related to TravelCCM.

2 TravelCCM People

2.1 Project Admins

- Denis Arnaud <`denis_arnaud@users.sourceforge.net`> (N)

- Anh Quan Nguyen <`quannaus@users.sourceforge.net`> (N)

2.2 Developers

- Anh Quan Nguyen <`quannaus@users.sourceforge.net`> (N)

- Denis Arnaud <`denis_arnaud@users.sourceforge.net`> (N)

- Gabrielle Sabatier <`gsabatier@users.sourceforge.net`> (N)

2.3 Retired Developers

- Benoit Lardeux <`benlardeux@users.sourceforge.net`> (N)

- Mehdi Ayouni <`mehdi.ayouni@gmail.com`>

- Alexandre Point <`apoint@users.sourceforge.net`>

2.4 Contributors

- Dongkyoung Choe <`dchoe@users.sourceforge.net`> (N)

- Emmanuel Bastien <ebastien@users.sourceforge.net> (N)

- Christophe Lacombe <ddtof@users.sourceforge.net> (N)

2.5   Distribution Maintainers

- Fedora/RedHat: Denis Arnaud <denis_arnaud@users.sourceforge.net> (N)

- Debian: Emmanuel Bastien <ebastien@users.sourceforge.net> (N)

Note

(N) - Amadeus employees.

3   Coding Rules

In the following sections we describe the naming conventions which are used for files, classes, structures, local variables, and global variables.

3.1   Default Naming Rules for Variables

Variables are named using lower-case letters and words are separated using underscore. Abbreviations, when used in variable names, are also written with lower-case letters. Examples:

- `fft_size`

- `nrof_paths`

- `my_variable_name`

Some variable names or parts of variable names are commonly used in several different functions and files to denote the same thing. For instance the following common names and prefixes should be used:

- `rows` - number of rows in a matrix

- `cols` - number of columns in a matrix

- `nrof_` - number of ...

3.2   Default Naming Rules for Files

Files are named using lower-case letters and words are separated using under-score. Abbreviations, when used in file names, are also written with lower-case letters.

Source files are named using '`.cpp`' suffix, whereas header files end with '`.h`' extension. Examples:

- '`my_file.h`'

- '`my_file.cpp`'

3.3   Default Naming Rules for Functions

Function names are named using lower-case letters and words are separated using under-score. Abbreviations, when used in function names, are also written with lower-case letters. This rule applies both to stand-alone functions as well as to member functions of classes. Example:

- `int my_function_name(int a, int b)`

3.4   Default Naming Rules for Classes and Structures

Each new word in a class or structure name should always start with a capital letter and the words should be separated with an under-score. Abbreviations are written with capital letters. Examples:

- '`My_Class_Name`'

- '`My_Struct_Name`'

- '`OFDM`'

3.5   Default Functionality of Classes

All classes that are configured by input parameters should include:

- default empty constructor

- one or more additional constructor(s) that takes input parameters and initializes the class instance

- setup function, preferably named '`setup`' or '`set_parameters`'

Explicit destructor functions are not required, unless they are needed. It shall not be possible to use any of the other member functions unless the class has been properly initiated with the input parameters.

4 Copyright and License

4.1 GNU LESSER GENERAL PUBLIC LICENSE

4.1.1 Version 2.1, February 1999

```
Copyright (C) 1991, 1999 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA  02110-1301  USA

Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL.  It also counts
as the successor of the GNU Library Public License, version 2, hence
the version number 2.1.]
```

4.2 Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's

reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

4.3   TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may

be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) The modified work must itself be a software library.

b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional:

if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the

Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However,

as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

4.3.1 NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

4.3.2 END OF TERMS AND CONDITIONS

4.4 How to Apply These Terms to Your New Programs

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

```
<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year>  <name of author>

This library is free software; you can redistribute it and/or
modify it under the terms of the GNU Lesser General Public
License as published by the Free Software Foundation; either
version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the GNU
Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public
License along with this library; if not, write to the Free Software
Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA  02110-1301  USA
```

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

```
Yoyodyne, Inc., hereby disclaims all copyright interest in the
library 'Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice
```

That's all there is to it!

Source

5 Documentation Rules

5.1 General Rules

All classes in IT++ should be properly documented with Doxygen comments in include ('.h') files. Source ('.cpp') files should be documented according to a normal standard for well documented C++ code.

An example of how the interface of a class shall be documented in IT++ is shown here:

```
/*!
 * \brief Brief description of My_Class here
 *
 * Detailed description of My_Class here. With example code if needed.
 */
class My_Class {

public:

  //! Default constructor
  My_Class(void) { setup_done = false; }

  /*!
   * \brief Constructor that initializes the class with parameters
   *
   * Detailed description of the constructor here if needed
   *
   * \param[in] param1 Description of \a param1 here
   * \param[in] param2 Description of \a param2 here
   */
  My_Class(TYPE1 param1, TYPE2 param2) { setup(param1, param2); }

  /*!
   * \brief Setup function for My_Class
   *
   * Detailed description of the setup function here if needed
   *
   * \param[in] param1 Description of \a param1 here
   * \param[in] param2 Description of \a param2 here
   */
  void setup(TYPE1 param1, TYPE2 param2);

  /*!
   * \brief Brief description of member_function1
   *
   * Detailed description of member_function1 here if needed
   *
   * \param[in]     param1 Description of \a param1 here
   * \param[in]     param2 Description of \a param2 here
   * \param[in,out] param3 Description of \a param3 here
   * \return Description of the return value here
   */
  TYPE4 member_function1(TYPE1 param1, TYPE2 param2, TYPE3 &param3);
```

```
private:

  bool setup_done;          /*!< Variable that checks if the class is properly
                                 initialized with parameters */
  TYPE1 private_variable1; //!< Short description of private_variable1 here
  TYPE2 private_variable2; //!< Short description of private_variable2 here
};
```

5.2 File Header

All files should start with the following header, which include Doxygen's `\file`, `\brief` and `\author` tags, `$Date$` and `$Revisions$` CVS tags, and a common copyright note:

```
/*!
 * \file
 * \brief Brief description of the file here
 * \author Names of the authors who contributed to this code
 *
 * Detailed description of the file here if needed.
 *
 * $Date: 2005-12-13 12:27:39 +0100 (mar, 13 déc 2005) $
 * $Revision: 93 $
 *
 * -------------------------------------------------------------------------
 *
 * IT++ - C++ library of mathematical, signal processing, speech processing,
 *        and communications classes and functions
 *
 * Copyright (C) 1995-2005  (see AUTHORS file for a list of contributors)
 *
 * This program is free software; you can redistribute it and/or modify
 * it under the terms of the GNU General Public License as published by
 * the Free Software Foundation; either version 2 of the License, or
 * (at your option) any later version.
 *
 * This program is distributed in the hope that it will be useful,
 * but WITHOUT ANY WARRANTY; without even the implied warranty of
 * MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
 * GNU General Public License for more details.
 *
 * You should have received a copy of the GNU General Public License
 * along with this program; if not, write to the Free Software
 * Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA
 *
 * -------------------------------------------------------------------------
 */
```

5.3 Grouping Various Parts

All functions must be added to a Doxygen group in order to appear in the documentation. The following code example defines the group 'my_group':

```
/*!
 * \defgroup my_group Brief description of the group here
 *
```

```
 * Detailed description of the group here
*/
```

The following example shows how to document the function `my_function` and how to add it to the group `my_group`:

```
/*!
 * \brief Brief description of my_function here
 * \ingroup my_group
 *
 * Detailed description of my_function here
 *
 * \param[in] param1 Description of \a param1 here
 * \param[in] param2 Description of \a param2 here
 * \return Description of the return value here
 */
TYPE3 my_function(TYPE1 param1, TYPE2 &param2);
```

6   Main features of TravelCCM

A short list of the main features of TravelCCM is given below sorted in different categories. Some more features and functions exist and for these we refer to the reference documentation.

6.1   Customer-Choice features

- Customer-Choice Model (CCM)

6.2   Other features

- CSV input file parsing

7   Make a Difference

Do not ask what TravelCCM can do for you. Ask what you can do for TravelCCM.

You can help us to develop the TravelCCM library. There are always a lot of things you can do:

- Start using TravelCCM

- Tell your friends about TravelCCM and help them to get started using it

- If you find a bug, report it to us. Without your help we can never hope to produce a bug free code.

- Help us to improve the documentation by providing information about documentation bugs

- Answer support requests in the TravelCCM discussion forums on SourceForge. If you know the answer to a question, help others to overcome their TravelCCM problems.

- Help us to improve our algorithms. If you know of a better way (e.g. that is faster or requires less memory) to implement some of our algorithms, then let us know.

- Help us to port TravelCCM to new platforms. If you manage to compile TravelCCM on a new platform, then tell us how you did it.

- Send us your code. If you have a good TravelCCM compatible code, which you can release under the LGPLv2.1, and you think it should be included in Travel-CCM, then send it to us.

- Become an TravelCCM developer. Send us an e-mail and tell what you can do for TravelCCM.

8   Make a new release

8.1   Introduction

This document describes briefly the recommended procedure of releasing a new version of TravelCCM using a Linux development machine and the SourceForge project site.

The following steps are required to make a release of the distribution package.

- Initialisation

- Release branch maintenance

- Commit and publish the release branch

- Create source packages (tar-balls)

- Upload the HTML documentation to SourceForge

- Generate the RPM packages

- Update distributed change log

- Create the binary package, including the documentation

- Upload the files to SourceForge

- Make a new post

- Send an email on the announcement mailing-list

8.2   Initialisation

Clone locally the full Git project:

```
cd ~
mkdir -p dev/sim
cd ~/dev/sim
git clone git://travel-ccm.git.sourceforge.net/gitroot/travel-ccm/travel-ccm travelccmgit
cd travelccmgit
git checkout trunk
```

8.3   Release branch maintenance

Switch to the release branch, on your local clone, and merge the latest updates from the trunk. Decide about the new version to be released.

```
cd ~/dev/sim/travelccmgit
git checkout releases
git merge trunk
```

Update the version in the various build system files, replacing the old version numbers by the correct ones:

```
vi CMakeLists.txt
vi autogen.sh
vi README
```

Update the version, add some news in the NEWS file, add a change-log in the ChangeLog file and in the RPM specification files:

```
vi NEWS
vi ChangeLog
vi travelccm.spec
```

8.4   Commit and publish the release branch

Commit the new release:

```
cd ~/dev/sim/travelccmgit
git add -A
git commit -m "[Release 0.5.0] Release of the 0.5.0 version of TravelCCM."
git push
```

8.5   Create source packages (tar-balls)

Create the distribution packages using the following command:

```
cd ~/dev/sim/travelccmgit
git checkout releases
rm -rf build && mkdir -p build
cd build
export INSTALL_BASEDIR=/home/user/dev/deliveries
export LIBSUFFIX_4_CMAKE="-DLIB_SUFFIX=64"
```

```
cmake -DCMAKE_INSTALL_PREFIX=${INSTALL_BASEDIR}/travelccm-0.5.0 \
 -DWITH_STDAIR_PREFIX=${INSTALL_BASEDIR}/stdair-stable \
 -DWITH_AIRRAC_PREFIX=${INSTALL_BASEDIR}/airsched-stable \
 -DWITH_AIRRAC_PREFIX=${INSTALL_BASEDIR}/airrac-stable \
 -DWITH_TravelCCM_PREFIX=${INSTALL_BASEDIR}/travelccm-stable \
 -DWITH_TravelCCM_PREFIX=${INSTALL_BASEDIR}/airinv-stable \
 -DWITH_TravelCCM_PREFIX=${INSTALL_BASEDIR}/simfqt-stable \
 -DCMAKE_BUILD_TYPE:STRING=Debug -DINSTALL_DOC:BOOL=ON \
 ${LIBSUFFIX_4_CMAKE} ..
make check && make dist
make install
```

This will configure, compile and check the package. The output packages will be named,
for instance, `travelccm-0.5.0.tar.gz` and `travelccm-0.5.0.tar.bz2`.

8.6   Upload the HTML documentation to SourceForge

In order to update the Web site files, either:

- `synchronise them with rsync and SSH`: Upload the just generated
  HTML (and PDF) documentation onto the `SourceForge Web site`.

  ```
  cd ~/dev/sim/travelccmgit/build
  git checkout releases
  rsync -aiv ${INSTALL_BASEDIR}/travelccm-0.5.0/share/doc/travelccm-0.5.0/html/ \
   your_sf_user,travel-ccm@web.sourceforge.net:htdocs/
  ```

  where `-aiv` options mean:

    - `-a`: archive/mirror mode; equals `-rlptgoD` (no `-H`, `-A`, `-X`)
    - `-v`: increase verbosity
    - `-i`: output a change-summary for all updates
    - Note the trailing slashes (/) at the end of both the source and target di-
      rectories. It means that the content of the source directory (`doc/html`),
      rather than the directory itself, has to be copied into the content of the target
      directory.

- or use the `SourceForge Shell service`.

8.7   Generate the RPM packages

Optionally, generate the RPM package (for instance, for `Fedora`/`RedHat`):

```
cd ~/dev/sim/travelccmgit/build
git checkout releases
make dist
```

To perform this step, rpm-build, rpmlint and rpmdevtools have to be available on the
system.

```
cp ../travelccm.spec ~/dev/packages/SPECS \
  && cp travelccm-0.5.0.tar.bz2 ~/dev/packages/SOURCES
cd ~/dev/packages/SPECS
rpmbuild -ba travelccm.spec
cd ~/dev/packages
rpmlint -i SPECS/travelccm.spec SRPMS/travelccm-0.5.0-1.fc16.src.rpm \
  RPMS/noarch/travelccm-* RPMS/i686/travelccm-*
```

8.8 Update distributed change log

Update the `NEWS` and `ChangeLog` files with appropriate information, including what has changed since the previous release. Then commit and push the changes into the `TravelCCM's Git repository`.

8.9 Create the binary package, including the documentation

Create the binary package, which includes HTML and PDF documentation, using the following command:

```
cd ~/dev/sim/travelccmgit/build
git checkout releases
make package
```

The output binary package will be named, for instance, `travelccm-0.5.0-Linux.tar.bz2`. That package contains both the HTML and PDF documentation. The binary package contains also the executables and shared libraries, as well as C++ header files, but all of those do not interest us for now.

8.10 Upload the files to SourceForge

Upload the distribution and documentation packages to the SourceForge server. Check `SourceForge help page on uploading software`.

8.11 Make a new post

- submit a new entry in the `SourceForge project-related news feed`
- make a new post on the `SourceForge hosted WordPress blog`
- and update, if necessary, `Trac tickets`.

8.12 Send an email on the announcement mailing-list

Finally, you should send an announcement to `travelccm-announce@lists.sourceforge.net` (see `https://lists.sourceforge.net/lists/listinfo/travelccm-announce` for the archives)

9 Installation

9.1 Table of Contents

9.2 Fedora/RedHat Linux distributions

Note that on Fedora/RedHat Linux distributions, RPM packages are available and can be installed with your usual package manager. For instance:

```
yum -y install travelccm-devel travelccm-doc
```

RPM packages can also be available on the SourceForge download site.

9.3 TravelCCM Requirements

TravelCCM should compile without errors or warnings on most GNU/Linux systems, on UNIX systems like Solaris SunOS, and on POSIX based environments for Microsoft Windows like Cygwin or MinGW with MSYS. It can be also built on Microsoft Windows NT/2000/XP/Vista/7 using Microsoft's Visual C++ .NET, but our support for this compiler is limited. For GNU/Linux, SunOS, Cygwin and MinGW we assume that you have at least the following GNU software installed on your computer:

- GNU Autotools:

    - autoconf,

    - automake,

    - libtool,

- – `make`, version 3.72.1 or later (check version with '`make` --version')

- `GCC` - GNU C++ Compiler (g++), version 4.3.x or later (check version with '`gcc`
  --version')

- `Boost` - C++ STL extensions, version 1.35 or later (check version with '`grep`
  "define BOOST_LIB_VERSION" /usr/include/boost/version.hpp')

- `MySQL` - Database client libraries, version 5.0 or later (check version with '`mysql`
  --version')

- `SOCI` - C++ database client library wrapper, version 3.0.0 or later (check version
  with '`soci-config` --version')

Optionally, you might need a few additional programs: `Doxygen`, `LaTeX`, `Dvips` and
`Ghostscript`, to generate the HTML and PDF documentation.

We strongly recommend that you use recent stable releases of the GCC, if possible. We
do not actively work on supporting older versions of the GCC, and they may therefore
(without prior notice) become unsupported in future releases of TravelCCM.

9.4   Basic Installation

Briefly, the shell commands '`./cmake ..  && make install`' should config-
ure, build, and install this package. The following more-detailed instructions are generic;
see the '`README`' file for instructions specific to this package. Some packages pro-
vide this '`INSTALL`' file but do not implement all of the features documented below.
The lack of an optional feature in a given package is not necessarily a bug. More recom-
mendations for GNU packages can be found in the info page corresponding to "Makefile
Conventions: (standards)Makefile Conventions".

The '`cmake`' shell script attempts to guess correct values for various system-dependent
variables used during compilation. It uses those values to create a '`Makefile`' in
each directory of the package. It may also create one or more '.h' files containing
system-dependent definitions. Finally, it creates a '`CMakeCache`.txt' cache file that
you can refer to in the future to recreate the current configuration, and a file '`CMakeFiles`'
containing compiler output (useful mainly for debugging '`cmake`').

It can also use an optional file (typically called '`config`.cache' and enabled with
'`--cache-file=config`.cache' or simply '`-C`') that saves the results of its tests
to speed up reconfiguring. Caching is disabled by default to prevent problems with
accidental use of stale cache files.

If you need to do unusual things to compile the package, please try to figure out how
'configure' could check whether to do them, and mail diffs or instructions to the address
given in the '`README`' so they can be considered for the next release. If you are using
the cache, and at some point '`config`.cache' contains results you don't want to keep,
you may remove or edit it.

The file '`CMakeLists.txt`' is used to create the '`Makefile`' files.

The simplest way to compile this package is:

1. `cd` to the directory containing the package's source code and type `./cmake ..` to configure the package for your system. Running `cmake` is generally fast. While running, it prints some messages telling which features it is checking for.

2. Type `make` to compile the package.

3. Optionally, type `make check` to run any self-tests that come with the package, generally using the just-built uninstalled binaries.

4. Type `make install` to install the programs and any data files and documentation. When installing into a prefix owned by root, it is recommended that the package be configured and built as a regular user, and only the `make install` phase executed with root privileges.

5. You can remove the program binaries and object files from the source code directory by typing `make clean`. To also remove the files that `configure` created (so you can compile the package for a different kind of computer), type `make distclean`. There is also a `make maintainer-clean` target, but that is intended mainly for the package's developers. If you use it, you may have to get all sorts of other programs in order to regenerate files that came with the distribution.

6. Often, you can also type `make uninstall` to remove the installed files again. In practice, not all packages have tested that uninstallation works correctly, even though it is required by the GNU Coding Standards.

9.5 Compilers and Options

Some systems require unusual options for compilation or linking that the `cmake` script does not know about. Run `./cmake --help` for details on some of the pertinent environment variables.

You can give `cmake` initial values for configuration parameters by setting variables in the command line or in the environment. Here is an example:

```
./cmake  CC=c99 CFLAGS=-g LIBS=-lposix
```

See also

　　Defining Variables for more details.

9.6 Compiling For Multiple Architectures

You can compile the package for more than one kind of computer at the same time, by placing the object files for each architecture

in their own directory. To do this, you can use GNU 'make'.
'cd' to the directory where you want the object files and executables
to go and run the 'configure' script. 'configure' automatically
checks for the source code in the directory that 'configure'
is in and in '..'. This is known as a "VPATH" build.

With a non-GNU 'make', it is safer to compile the package for
one architecture at a time in the source code directory. After
you have installed the package for one architecture, use 'make
distclean' before reconfiguring for another architecture.

On MacOS X 10.5 and later systems, you can create libraries
and executables that work on multiple system types--known as
"fat" or "universal" binaries--by specifying multiple '-arch'
options to the compiler but only a single '-arch' option to
the preprocessor. Like this:

```
./configure CC="gcc -arch i386 -arch x86_64 -arch ppc -arch ppc64" \
            CXX="g++ -arch i386 -arch x86_64 -arch ppc -arch ppc64" \
            CPP="gcc -E" CXXCPP="g++ -E"
```

This is not guaranteed to produce working output in all cases,
you may have to build one architecture at a time and combine
the results using the 'lipo' tool if you have problems.

9.7 Installation Names

By default, 'make install' installs the package's commands
under '/usr/local/bin', include files under '/usr/local/include',
etc. You can specify an installation prefix other than '/usr/local'
by giving 'configure' the option '--prefix=PREFIX', where PREFIX
must be an absolute file name.

You can specify separate installation prefixes for architecture-specific
files and architecture-independent files. If you pass the
option '--exec-prefix=PREFIX' to 'configure', the package uses
PREFIX as the prefix for installing programs and libraries.
Documentation and other data files still use the regular prefix.

In addition, if you use an unusual directory layout you can
give options like '--bindir=DIR' to specify different values
for particular kinds of files. Run 'configure --help' for
a list of the directories you can set and what kinds of files
go in them. In general, the default for these options is expressed
in terms of '${prefix}', so that specifying just '--prefix'
will affect all of the other directory specifications that
were not explicitly provided.

The most portable way to affect installation locations is to
pass the correct locations to 'configure'; however, many packages
provide one or both of the following shortcuts of passing variable

assignments to the 'make install' command line to change installation
locations without having to reconfigure or recompile.

The first method involves providing an override variable for
each affected directory. For example, 'make install prefix=/alternate/director
will choose an alternate location for all directory configuration
variables that were expressed in terms of '${prefix}'. Any
directories that were specified during 'configure', but not
in terms of '${prefix}', must each be overridden at install
time for the entire installation to be relocated. The approach
of makefile variable overrides for each directory variable
is required by the GNU Coding Standards, and ideally causes
no recompilation. However, some platforms have known limitations
with the semantics of shared libraries that end up requiring
recompilation when using this method, particularly noticeable
in packages that use GNU Libtool.

The second method involves providing the 'DESTDIR' variable.
For example, 'make install DESTDIR=/alternate/directory' will
prepend '/alternate/directory' before all installation names.
The approach of 'DESTDIR' overrides is not required by the
GNU Coding Standards, and does not work on platforms that have
drive letters. On the other hand, it does better at avoiding
recompilation issues, and works well even when some directory
options were not specified in terms of '${prefix}' at 'configure'
time.

9.8 Optional Features

If the package supports it, you can cause programs to be installed
with an extra prefix or suffix on their names by giving 'cmake'
the option '--program-prefix=PREFIX' or '--program-suffix=SUFFIX'.

Some packages pay attention to '--enable-FEATURE' options to
'configure', where FEATURE indicates an optional part of the
package. They may also pay attention to '--with-PACKAGE' options,
where PACKAGE is something like 'gnu-as' or 'x' (for the X
Window System). The 'README' should mention any '--enable-'
and '--with-' options that the package recognizes.

For packages that use the X Window System, 'configure' can
usually find the X include and library files automatically,
but if it doesn't, you can use the 'configure' options '--x-includes=DIR'
and '--x-libraries=DIR' to specify their locations.

Some packages offer the ability to configure how verbose the
execution of 'make' will be. For these packages, running './configure
--enable-silent-rules' sets the default to minimal output,
which can be overridden with 'make V=1'; while running './configure
--disable-silent-rules' sets the default to verbose, which
can be overridden with 'make V=0'.

9.9 Particular systems

On HP-UX, the default C compiler is not ANSI C compatible.
If GNU CC is not installed, it is recommended to use the following
options in order to use an ANSI C compiler:

```
./configure CC="cc -Ae -D_XOPEN_SOURCE=500"
```

and if that doesn't work, install pre-built binaries of GCC
for HP-UX.

On OSF/1 a.k.a. Tru64, some versions of the default C compiler
cannot parse its '<wchar.h>' header file. The option '-nodtk'
can be used as a workaround. If GNU CC is not installed, it
is therefore recommended to try

```
./configure CC="cc"
```

and if that doesn't work, try

```
./configure CC="cc -nodtk"
```

On Solaris, don't put '/usr/ucb' early in your 'PATH'. This
directory contains several dysfunctional programs; working
variants of these programs are available in '/usr/bin'. So,
if you need '/usr/ucb' in your 'PATH', put it _after_ '/usr/bin'.

On Haiku, software installed for all users goes in '/boot/common',
not '/usr/local'. It is recommended to use the following options:

```
./cmake -DCMAKE_INSTALL_PREFIX=/boot/common
```

9.10 Specifying the System Type

There may be some features 'configure' cannot figure out automatically,
but needs to determine by the type of machine the package will
run on. Usually, assuming the package is built to be run on
the _same_ architectures, 'configure' can figure that out,
but if it prints a message saying it cannot guess the machine
type, give it the '--build=TYPE' option. TYPE can either be
a short name for the system type, such as 'sun4', or a canonical
name which has the form CPU-COMPANY-SYSTEM

where SYSTEM can have one of these forms:

- OS

- KERNEL-OS

See the file 'config.sub' for the possible values of each field.
If 'config.sub' isn't included in this package, then this package
doesn't need to know the machine type.

If you are _building_ compiler tools for cross-compiling, you
should use the option '--target=TYPE' to select the type of
system they will produce code for.

If you want to _use_ a cross compiler, that generates code
for a platform different from the build platform, you should
specify the "host" platform (i.e., that on which the generated
programs will eventually be run) with '--host=TYPE'.

9.11 Sharing Defaults

If you want to set default values for 'configure' scripts to
share, you can create a site shell script called 'config.site'
that gives default values for variables like 'CC', 'cache_-
file', and 'prefix'. 'configure' looks for 'PREFIX/share/config.site'
if it exists, then 'PREFIX/etc/config.site' if it exists. Or,
you can set the 'CONFIG_SITE' environment variable to the location
of the site script. A warning: not all 'configure' scripts
look for a site script.

9.12 Defining Variables

Variables not defined in a site shell script can be set in
the environment passed to 'configure'. However, some packages
may run configure again during the build, and the customized
values of these variables may be lost. In order to avoid this
problem, you should set them in the 'configure' command line,
using 'VAR=value'. For example:

```
./configure CC=/usr/local2/bin/gcc
```

causes the specified 'gcc' to be used as the C compiler (unless
it is overridden in the site shell script).

Unfortunately, this technique does not work for 'CONFIG_SHELL'
due to an Autoconf bug. Until the bug is fixed you can use
this workaround:

```
CONFIG_SHELL=/bin/bash /bin/bash ./configure CONFIG_SHELL=/bin/bash
```

9.13 'cmake' Invocation

'cmake' recognizes the following options to control how it
operates.

- '--help', '-h' print a summary of all of the options to
  'cmake', and exit.

- '--help=short', '--help=recursive' print a summary of the
  options unique to this package's 'configure', and exit.
  The 'short' variant lists options used only in the top
  level, while the 'recursive' variant lists options also
  present in any nested packages.

- '--version', '-V' print the version of Autoconf used to
  generate the 'configure' script, and exit.

- '--cache-file=FILE' enable the cache: use and save the
  results of the tests in FILE, traditionally 'config.cache'.
  FILE defaults to '/dev/null' to disable caching.

- '--config-cache', '-C' alias for '--cache-file=config.cache'.

- '--quiet', '--silent', '-q' do not print messages saying
  which checks are being made. To suppress all normal output,
  redirect it to '/dev/null' (any error messages will still
  be shown).

- '--srcdir=DIR' look for the package's source code in directory
  DIR. Usually 'configure' can determine that directory automatically.

- '--prefix=DIR' use DIR as the installation prefix.

  See also

  > Installation Names for more details, including other
  > options available for fine-tuning the installation
  > locations.

- '--no-create', '-n' run the configure checks, but stop
  before creating any output files.

'cmake' also accepts some other, not widely useful, options.
Run 'cmake' --help' for more details.

The 'cmake' script produces an ouput like this:

```
-- Requires Git without specifying any version
cmake -DCMAKE_INSTALL_PREFIX=/home/user/dev/deliveries/travelccm-99.99.99 -DLIB_SUFFIX=64 -DCMAKE_
-- The C compiler identification is GNU
-- The CXX compiler identification is GNU
-- Check for working C compiler: /usr/lib64/ccache/gcc
-- Check for working C compiler: /usr/lib64/ccache/gcc -- works
-- Detecting C compiler ABI info
-- Detecting C compiler ABI info - done
-- Check for working CXX compiler: /usr/lib64/ccache/c++
-- Check for working CXX compiler: /usr/lib64/ccache/c++ -- works
-- Detecting CXX compiler ABI info
-- Detecting CXX compiler ABI info - done
-- Requires Git without specifying any version
-- Current Git revision name: dbedfae29cd555c176034d09ecff12e80f28e915 trunk
-- Requires Boost-1.41
-- Boost version: 1.46.0
-- Found the following Boost libraries:
--   regex
```

```
--    program_options
--    date_time
--    iostreams
--    serialization
--    filesystem
--    unit_test_framework
--    python
-- Found Boost version: 1.46.0
-- Found BoostWrapper: /usr/include (Required is at least version "1.41")
-- Requires MySQL without specifying any version
-- Using mysql-config: /usr/bin/mysql_config
-- Found MySQL: /usr/lib64/mysql/libmysqlclient.so
-- Found MySQL version: 5.5.14
-- Requires SOCI-3.0
-- Using soci-config: /usr/bin/soci-config
-- SOCI headers are buried
-- Found SOCI: /usr/lib64/libsoci_core.so (Required is at least version "3.0")
-- Found SOCIMySQL: /usr/lib64/libsoci_mysql.so (Required is at least version "3.0")
-- Found SOCI with MySQL back-end support version: 3.0.0
-- Requires StdAir-0.35
-- Found StdAir version: 0.38.0
-- Requires Doxygen without specifying any version
-- Found Doxygen: /usr/bin/doxygen
-- Found DoxygenWrapper: /usr/bin/doxygen
-- Found Doxygen version: 1.7.4
-- Had to set the linker language for 'travelccmlib' to CXX
-- Test 'TravelChoiceTestSuite' to be built with 'TravelChoiceTestSuite.cpp'
--
-- ==============================================================
-- --------------------------------
-- ---     Project Information    ---
-- --------------------------------
-- PROJECT_NAME ................... : travelccm
-- PACKAGE_PRETTY_NAME ............ : TravelCCM
-- PACKAGE ....................... : travelccm
-- PACKAGE_NAME ................... : TRAVELCCM
-- PACKAGE_BRIEF ................. : C++ Travel Customer Choice Model Library
-- PACKAGE_VERSION ............... : 99.99.99
-- GENERIC_LIB_VERSION ........... : 99.99.99
-- GENERIC_LIB_SOVERSION ......... : 99.99
--
-- --------------------------------
-- ---      Build Configuration    ---
-- --------------------------------
-- Modules to build ............... : travelccm
-- Libraries to build/install ..... : travelccmlib
-- Binaries to build/install ...... : travelccm
-- Modules to test ............... : travelccm
-- Binaries to test .............. : TravelChoiceTestSuitetst
--
-- * Module ...................... : travelccm
--   + Layers to build ........... : .;basic;bom;factory;command;service
--   + Dependencies on other layers :
--   + Libraries to build/install . : travelccmlib
--   + Executables to build/install : travelccm
--   + Tests to perform .......... : TravelChoiceTestSuitetst
--
-- BUILD_SHARED_LIBS ............. : ON
-- CMAKE_BUILD_TYPE .............. : Debug
--  * CMAKE_C_FLAGS ............... :
--  * CMAKE_CXX_FLAGS ............. : -Wall -Werror
--  * BUILD_FLAGS ................. :
```

```
--  * COMPILE_FLAGS ............... :
-- CMAKE_MODULE_PATH .............. : /home/user/dev/sim/travelccm/travelccmgithub/config/
-- CMAKE_INSTALL_PREFIX ........... : /home/user/dev/deliveries/travelccm-99.99.99
--
-- * Doxygen:
--   - DOXYGEN_VERSION .............. : 1.7.4
--   - DOXYGEN_EXECUTABLE ........... : /usr/bin/doxygen
--   - DOXYGEN_DOT_EXECUTABLE ....... : /usr/bin/dot
--   - DOXYGEN_DOT_PATH ............. : /usr/bin
--
-- ---------------------------------
-- --- Installation Configuration ---
-- ---------------------------------
-- INSTALL_LIB_DIR ................ : /home/user/dev/deliveries/travelccm-99.99.99/lib64
-- INSTALL_BIN_DIR ................ : /home/user/dev/deliveries/travelccm-99.99.99/bin
-- INSTALL_INCLUDE_DIR ............ : /home/user/dev/deliveries/travelccm-99.99.99/include
-- INSTALL_DATA_DIR ............... : /home/user/dev/deliveries/travelccm-99.99.99/share
-- INSTALL_SAMPLE_DIR ............. : /home/user/dev/deliveries/travelccm-99.99.99/share/travelccm,
-- INSTALL_DOC .................... : ON
--
-- ---------------------------------
-- ---   Packaging Configuration  ---
-- ---------------------------------
-- CPACK_PACKAGE_CONTACT .......... : Denis Arnaud <denis_arnaud - at - users dot sourceforge dot r
-- CPACK_PACKAGE_VENDOR ........... : Denis Arnaud
-- CPACK_PACKAGE_VERSION .......... : 99.99.99
-- CPACK_PACKAGE_DESCRIPTION_FILE . : /home/user/dev/sim/travelccm/travelccmgithub/README
-- CPACK_RESOURCE_FILE_LICENSE .... : /home/user/dev/sim/travelccm/travelccmgithub/COPYING
-- CPACK_GENERATOR ................ : TBZ2
-- CPACK_DEBIAN_PACKAGE_DEPENDS ... :
-- CPACK_SOURCE_GENERATOR ......... : TBZ2;TGZ
-- CPACK_SOURCE_PACKAGE_FILE_NAME . : travelccm-99.99.99
--
-- ---------------------------------
-- ---      External libraries   ---
-- ---------------------------------
--
-- * Boost:
--   - Boost_VERSION .............. : 104600
--   - Boost_LIB_VERSION .......... : 1_46
--   - Boost_HUMAN_VERSION ........ : 1.46.0
--   - Boost_INCLUDE_DIRS ......... : /usr/include
--   - Boost required components .. : regex;program_options;date_time;iostreams;serialization;files
--   - Boost required libraries ... : optimized;/usr/lib64/libboost_regex-mt.so;debug;/usr/lib64/li
--
-- * MySQL:
--   - MYSQL_VERSION .............. : 5.5.14
--   - MYSQL_INCLUDE_DIR .......... : /usr/include/mysql
--   - MYSQL_LIBRARIES ............ : /usr/lib64/mysql/libmysqlclient.so
--
-- * SOCI:
--   - SOCI_VERSION ............... : 3.0.0
--   - SOCI_INCLUDE_DIR ........... : /usr/include/soci
--   - SOCIMYSQL_INCLUDE_DIR ...... : /usr/include/soci
--   - SOCI_LIBRARIES ............. : /usr/lib64/libsoci_core.so
--   - SOCIMYSQL_LIBRARIES ........ : /usr/lib64/libsoci_mysql.so
--
-- * StdAir:
--   - STDAIR_VERSION ............. : 0.38.0
--   - STDAIR_BINARY_DIRS ......... : /home/user/dev/deliveries/stdair-0.38.0/bin
--   - STDAIR_EXECUTABLES ......... : stdair
--   - STDAIR_LIBRARY_DIRS ........ : /home/user/dev/deliveries/stdair-0.38.0/lib64
```

```
--   - STDAIR_LIBRARIES ........... : stdairlib;stdairuicllib
--   - STDAIR_INCLUDE_DIRS ........ : /home/user/dev/deliveries/stdair-0.38.0/include
--   - STDAIR_SAMPLE_DIR .......... : /home/user/dev/deliveries/stdair-0.38.0/share/stdair/samples
--
-- Change a value with: cmake -D<Variable>=<Value>
-- ============================================================
--
-- Configuring done
-- Generating done
-- Build files have been written to: /home/user/dev/sim/travelccm/travelccmgithub/build
```

It is recommended that you check if your library has been compiled and linked properly and works as expected. To do so, you should execute the testing process 'make check'. As a result, you should obtain a similar report:

```
[  0%] Built target hdr_cfg_travelccm
[ 88%] Built target travelccmlib
[100%] Built target TravelChoiceTestSuitetst
Scanning dependencies of target check_travelccmtst
Test project /home/user/dev/sim/travelccm/travelccmgithub/build/test/travelccm
    Start 1: TravelChoiceTestSuitetst
1/1 Test #1: TravelChoiceTestSuitetst .........   Passed    0.02 sec

100% tests passed, 0 tests failed out of 1

Total Test time (real) =   0.30 sec
[100%] Built target check_travelccmtst
Scanning dependencies of target check
[100%] Built target check
```

Check if all the executed tests PASSED. If not, please contact us by filling a bug-report.

Finally, you should install the compiled and linked library, include files and (optionally) HTML and PDF documentation by typing:

```
make install
```

Depending on the PREFIX settings during configuration, you might need the root (administrator) access to perform this step.

Eventually, you might invoke the following command

```
make clean
```

to remove all files created during compilation process, or even

```
cd ~/dev/sim/travelccmgit
rm -rf build && mkdir build
cd build
```

to remove everything.

10 Linking with TravelCCM

10.1   Table of Contents

10.2   Introduction

There are two convenient methods of linking your programs with the TravelCCM library. The first one employs the `pkg-config` command (see http://pkgconfig.freedesktop.org/), whereas the second one uses `travelccm-config` script. These methods are shortly described below.

10.3   Dependencies

The TravelCCM library depends on several other C++ components.

10.3.1   StdAir

Among them, as for now, only StdAir has been packaged. The support for StdAir is taken in charge by a dedicated M4 macro file (namely, `stdair.m4`), from the configuration script (generated thanks to `configure.ac`).
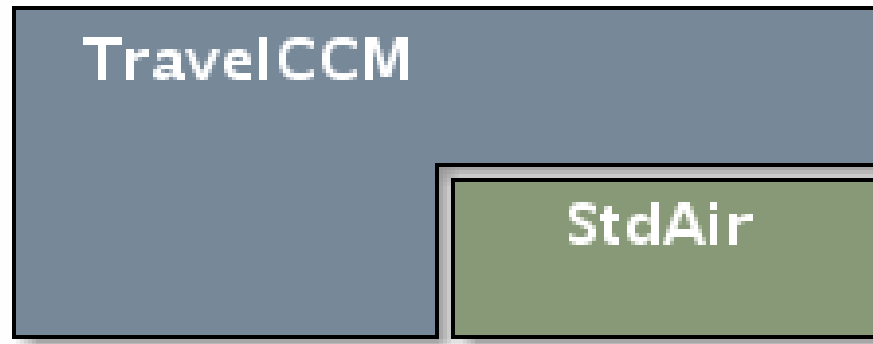
Figure 1: TravelCCM Dependencies

10.4   Using the pkg-config command

`pkg-config` is a helper tool used when compiling applications and libraries. It helps you insert the correct compiler and linker options. The syntax of the `pkg-config` is as follows:

```
pkg-config <options> <library_name>
```

For instance, assuming that you need to compile an TravelCCM based program `my_-prog.cpp`, you should use the following command:

```
g++ `pkg-config --cflags travelccm` -o my_prog my_prog.cpp `pkg-config --libs tra
    velccm`
```

For more information see the `pkg-config` man pages.

10.5   Using the travelccm-config script

TravelCCM provides a shell script called `travelccm-config`, which is installed by default in `$prefix/bin` (`/usr/local/bin`) directory. It can be used to simplify compilation and linking of TravelCCM based programs. The usage of this script is quite similar to the usage of the `pkg-config` command.

Assuming that you need to compile the program `my_prog.cpp` you can now do that with the following command:

```
g++ `travelccm-config --cflags` -o my_prog_opt my_prog.cpp `travelccm --li
    bs`
```

A list of `travelccm-config` options can be obtained by typing:

```
travelccm-config --help
```

If the `travelccm-config` command is not found by your shell, you should add its location `$prefix/bin` to the PATH environment variable, e.g.:

```
export PATH=/usr/local/bin:$PATH
```

10.6   M4 macro for the GNU Autotools

A M4 macro file is delivered with TravelCCM, namely 'travelccm.m4', which can be found in, e.g., '/usr/share/aclocal'. When used by a 'configure' script, thanks to he `AM_-PATH_TravelCCM` macro (specified in the M4 macro file), the following Makefile variables are then defined:

- `TravelCCM_VERSION` (e.g., defined to 0.23.0)

- `TravelCCM_CFLAGS` (e.g., defined to `-I${prefix}/include`)

- `TravelCCM_LIBS` (e.g., defined to `-L${prefix}/lib -ltravelccm`)

10.7   Using TravelCCM with dynamic linking

When using static linking some of the library routines in TravelCCM are copied into your executable program. This can lead to unnecessary large executables. To avoid having too large executable files you may use dynamic linking instead. Dynamic linking means that the actual linking is performed when the program is executed. This requires that the system is able to locate the shared TravelCCM library file during your program execution. If you install the TravelCCM library using a non-standard prefix, the `LD_LIBRARY_-PATH` environment variable might be used to inform the linker of the dynamic library location, e.g.:

```
export LD_LIBRARY_PATH=<TravelCCM installation prefix>/lib:$LD_LIBRARY_PATH
```

11   Test Rules

This section describes rules how the functionality of the IT++ library should be verified. In the `tests` subdirectory test files are provided. All functionality should be tested using these test files.

11.1 The Test File

Each new IT++ module/class should be accompanied with a test file. The test file is an implementation in C++ that tests the functionality of a function/class or a group of functions/classes called modules. The test file should test relevant parameter settings and input/output relations to guarantee correct functionality of the corresponding classes/functions. The test files should be maintained using version control and updated whenever new functionality is added to the IT++ library.

The test file should print relevant data to a standard output that can be used to verify the functionality. All relevant parameter settings should be tested.

The test file should be placed in the `'tests'` subdirectory and should have a name ending with `'_test.cpp'`.

11.2 The Reference File

Consider a test file named `'module_test.cpp'`. A reference file named `'module_-test.ref'` should accompany the test file. The reference file contains a reference printout of the standard output generated when running the test program. The reference file should be maintained using version control and updated according to the test file.

11.3 Testing IT++ Library

One can compile and execute all test programs from `'tests'` subdirectory by typing

```
% make check
```

after successful compilation of the IT++ library.

12 Users Guide

12.1 Table of Contents

- Introduction
- Get Started
    - Get the TravelCCM library
    - Build the TravelCCM project
    - Build and Run the Tests
    - Install the TravelCCM Project (Binaries, Documentation)
- Input file of TravelCCM Project
- The schedule BOM Tree
    - Build of the schedule BOM tree

– [Display of the schedule BOM tree](#)

- [Exploring the Predefined BOM Tree](#)

  – [Airline Network BOM Tree](#)

  – [Airline Schedule BOM Tree](#)

- [Extending the BOM Tree](#)

- [The travel solution calculation procedure](#)

12.2 Introduction

The `TravelCCM` library contains classes for airline business management. This document does not cover all the aspects of the `TravelCCM` library. It does however explain the most important things you need to know in order to start using `TravelCCM`.

12.3 Get Started

12.3.1 Get the TravelCCM library

Clone locally the full `Git project`:

```
cd ~
mkdir -p dev/sim
cd ~/dev/sim
git clone git://travel-ccm.git.sourceforge.net/gitroot/travel-ccm/travel-ccm travelccmgit
cd travelccmgit
git checkout trunk
```

12.3.2 Build the TravelCCM project

Link with StdAir, create the distribution package (say, 0.5.0) and compile using the following commands:

```
cd ~/dev/sim/travelccmgit
rm -rf build && mkdir -p build
cd build
cmake -DCMAKE_INSTALL_PREFIX=~/dev/deliveries/travelccm-0.5.0 \
 -DWITH_STDAIR_PREFIX=~/dev/deliveries/stdair-stable \
 -DCMAKE_BUILD_TYPE:STRING=Debug -DINSTALL_DOC:BOOL=ON ..
make
```

12.3.3 Build and Run the Tests

After building the TravelCCM project, the following commands run the tests:

```
cd ~/dev/sim/travelccmgit
cd build
make check
```

As a result, you should obtain a similar report:

```
[  0%] Built target hdr_cfg_travelccm
[ 96%] Built target travelccmlib
[100%] Built target AirlineScheduleTestSuitetst
Scanning dependencies of target check_travelccmtst
Test project /home/user/dev/sim/travelccm/travelccmgithub/build/test/travelccm
    Start 1: AirlineScheduleTestSuitetst
1/1 Test #1: AirlineScheduleTestSuitetst ......   Passed    0.15 sec

100% tests passed, 0 tests failed out of 1

Total Test time (real) =   0.40 sec
[100%] Built target check_travelccmtst
Scanning dependencies of target check
[100%] Built target check
```

12.3.4   Install the TravelCCM Project (Binaries, Documentation)

After the step Build the TravelCCM project, to install the library and its header files, type:

```
cd ~/dev/sim/travelccmgit
cd build
make install
```

You can check that the executables and other required files have been copied into the given final directory:

```
cd ~dev/deliveries/travelccm-0.5.0
```

To generate the TravelCCM project documentation, the commands are:

```
cd ~/dev/sim/travelccmgit
cd build
make doc
```

The TravelCCM project documentation is available in the following formats: HTML, LaTeX. Those documents are available in a subdirectory:

```
cd ~/dev/sim/travelccmgit
cd build
cd doc
```

12.4   Input file of TravelCCM Project

The schedule input file structure should look like the following sample:

Each line, beyond the header, represents a schedule entry, i.e., the specification of a given flight-period (see `TRAVELCCM::FlightPeriodStruct`). The fields are as follows:

- Flights section

    - AirlineCode (e.g., `BA`)
    - FlightNumber (e.g., `9`)
    - Start of the flight departure period (e.g., `2007-04-20`)
    - End of the flight departure period (e.g., `2007-06-30`)
    - Day-Of-the-Week for the flight departure period (DOW) (e.g., `0000011`)
    - Leg section
    - Segment section

- Leg section

    - BoardPoint (e.g., `LHR`)
    - OffPoint (e.g., `BKK`)
    - BoardTime (e.g., `22`:00)
    - ArrivalTime (e.g., `15`:15)
    - ArrivalDateOffSet (e.g., `+1`)
    - ElapsedTime (e.g., `11`:15)
    - Leg-cabin section

- Leg-cabin section

    - Cabin code (e.g., `F`, `J`, `W` or `Y`)
    - Capacity (e.g., respectively `5`, `12`, `20` or `300`)

- Segment section

    - Specificity flag:
        * `0` means that all the segments behave the same way, i.e., have got the same dressing (distribution and order of the booking classes per cabin)
        * `1` means that each segment bahave differently. The full specification of each of those segments must therefore be given.
    - Segment-cabin section
    - Fare family section

- Segment-cabin section

    - Cabin code (e.g., `F`, `J`, `W` or `Y`)
    - List of (one-letter-code) booking classes for the cabin (e.g, respectively `FA`, `JCDI`, `WT` or `YBHKMLSQ`)

- Fare family section

    - Fare family code (e.g., `1`)
    - List of (one-letter-code) booking classes for the fare family (e.g, respectively `FA`, `JCDI`, `WT` or `YBHKMLSQ`)

Some fare input examples (including the example above named `schedule03.csv`) are given in the `StdAir project`.

12.5   The schedule BOM Tree

The schedule-related Business Object Model (BOM) tree is a structure allowing to store all the `TRAVELCCM::FlightPeriodStruct` objects of the simulation. That is why parsing an input file, containing the specification for all the flight-periods, is more convenient (

See also

> the previous section Input file of TravelCCM Project).

As it may be time consuming, and it for sure requires some know-how, to first build such a schedule input file, a small sample BOM tree is provided by default when needed.

12.5.1   Build of the schedule BOM tree

First, a BOM root object (i.e., a root for all the classes in the project) is instantiated by the `stdair::STDAIR_ServiceContext` context object, when the `stdair::STDAIR_-Service` is itself instantiated (during the instantiation of the `TRAVELCCM::TRAVELCCM_-Service` object).

The corresponding `type` (class) `stdair::BomRoot` is defined in the StdAir library.

Then, the BOM root can be either constructed thanks to the `TRAVELCCM::TRAVELCCM_-Service::buildSampleBom()` method:

```
void buildSampleBom();
```

or can be constructed using the schedule input file described above thanks to the `TRAVELCCM::TRAVELCCM_Service::parseAndLoad (const stdair::Filename_-T&)` method:

12.5.2   Display of the schedule BOM tree

Note

> That feature (of BOM tree display) has not been implemented yet. Do not hesitate to open a ticket if you would like to have it implemented more quickly.

The schedule BOM tree can be displayed as done in the `batches::travelccm.cpp` program:

When the default BOM tree is used (`-b/--builtin` option of the main program `travelccm.cpp`), the schedule BOM tree display (for now, corresponding to `schedule01.csv` parsed by `TRAVELCCM::travelccm`) should look like:

```
=================================================================
BomRoot:  -- ROOT --
=================================================================
+++++++++++++++++++++++++++++++++++++++++++++++++++
Inventory: SQ
+++++++++++++++++++++++++++++++++++++++++++++++++++
*******************************************
FlightDate: SQ11, 2010-Jan-15
*******************************************
*******************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
      apsed, Distance, Capacity,
SQ11 2010-Jan-15, SIN-BKK, 2010-Jan-15, 08:20:00, 2010-Jan-15, 11:00:00, 07:40:00
      , 0, -05:00:00, 6300, 0,
*******************************************
*******************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
      Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ11 2010-Jan-15, SIN-BKK 2010-Jan-15, Y, 300, 300, 0, 0, 0, 0, 0, 0, 0, 2, 298,
      9, 0, 0, 0, 0, 0,
*******************************************
*******************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ11 2010-Jan-15, SIN-BKK 2010-Jan-15, Y, 1, 0, 0, 0, 2, 298, 0,
SQ11 2010-Jan-15, SIN-BKK 2010-Jan-15, Y, 2, 0, 0, 0, 2, 298, 0,
*******************************************
*******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
      (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ11 2010-Jan-15, SIN-BKK 2010-Jan-15, Y, 1, Y, 300 (0), 0, 0, 0, 2, 0 (0), 0, 0,
      0, 0, 0, 0,
SQ11 2010-Jan-15, SIN-BKK 2010-Jan-15, Y, 2, M, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
*******************************************
*******************************************
FlightDate: SQ11, 2010-Jan-16
*******************************************
*******************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
      apsed, Distance, Capacity,
SQ11 2010-Jan-16, SIN-BKK, 2010-Jan-16, 08:20:00, 2010-Jan-16, 11:00:00, 07:40:00
      , 0, -05:00:00, 6300, 0,
*******************************************
*******************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
```

```
        Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ11 2010-Jan-16, SIN-BKK 2010-Jan-16, Y, 300, 300, 0, 0, 0, 0, 0, 0, 0, 0, 300,
      9, 1.83244e-319, 0, 0, 0, 0,
*******************************************
*******************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ11 2010-Jan-16, SIN-BKK 2010-Jan-16, Y, 1, 0, 0, 0, 0, 300, 0,
SQ11 2010-Jan-16, SIN-BKK 2010-Jan-16, Y, 2, 0, 0, 0, 0, 300, 0,
*******************************************
*******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
      (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ11 2010-Jan-16, SIN-BKK 2010-Jan-16, Y, 1, Y, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
SQ11 2010-Jan-16, SIN-BKK 2010-Jan-16, Y, 2, M, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
*******************************************
*******************************************
FlightDate: SQ11, 2010-Jan-17
*******************************************
*******************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
      apsed, Distance, Capacity,
SQ11 2010-Jan-17, SIN-BKK, 2010-Jan-17, 08:20:00, 2010-Jan-17, 11:00:00, 07:40:00
      , 0, -05:00:00, 6300, 0,
*******************************************
*******************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
      Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ11 2010-Jan-17, SIN-BKK 2010-Jan-17, Y, 300, 300, 0, 0, 0, 0, 0, 0, 0, 0, 300,
      9, 1.58896e-319, 0, 0, 0, 0,
*******************************************
*******************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ11 2010-Jan-17, SIN-BKK 2010-Jan-17, Y, 1, 0, 0, 0, 0, 300, 0,
SQ11 2010-Jan-17, SIN-BKK 2010-Jan-17, Y, 2, 0, 0, 0, 0, 300, 0,
*******************************************
*******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
      (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
```

```
SQ11 2010-Jan-17, SIN-BKK 2010-Jan-17, Y, 1, Y, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
SQ11 2010-Jan-17, SIN-BKK 2010-Jan-17, Y, 2, M, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
*******************************************
*******************************************
FlightDate: SQ11, 2010-Jan-18
*******************************************
*******************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
      apsed, Distance, Capacity,
SQ11 2010-Jan-18, SIN-BKK, 2010-Jan-18, 08:20:00, 2010-Jan-18, 11:00:00, 07:40:00
      , 0, -05:00:00, 6300, 0,
*******************************************
*******************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
      Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ11 2010-Jan-18, SIN-BKK 2010-Jan-18, Y, 300, 300, 0, 0, 0, 0, 0, 0, 0, 0, 300,
      9, 0, 0, 0, 0, 0,
*******************************************
*******************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ11 2010-Jan-18, SIN-BKK 2010-Jan-18, Y, 1, 0, 0, 0, 0, 300, 0,
SQ11 2010-Jan-18, SIN-BKK 2010-Jan-18, Y, 2, 0, 0, 0, 0, 300, 0,
*******************************************
*******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
      (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ11 2010-Jan-18, SIN-BKK 2010-Jan-18, Y, 1, Y, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
SQ11 2010-Jan-18, SIN-BKK 2010-Jan-18, Y, 2, M, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
*******************************************
*******************************************
FlightDate: SQ11, 2010-Jan-19
*******************************************
*******************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
      apsed, Distance, Capacity,
SQ11 2010-Jan-19, SIN-BKK, 2010-Jan-19, 08:20:00, 2010-Jan-19, 11:00:00, 07:40:00
      , 0, -05:00:00, 6300, 0,
*******************************************
*******************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
      Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
```

```
SQ11 2010-Jan-19, SIN-BKK 2010-Jan-19, Y, 300, 300, 0, 0, 0, 0, 0, 0, 0, 0, 300,
      9, 0, 0, 0, 0, 0,
*******************************************
*******************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ11 2010-Jan-19, SIN-BKK 2010-Jan-19, Y, 1, 0, 0, 0, 0, 300, 0,
SQ11 2010-Jan-19, SIN-BKK 2010-Jan-19, Y, 2, 0, 0, 0, 0, 300, 0,
*******************************************
*******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
      (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ11 2010-Jan-19, SIN-BKK 2010-Jan-19, Y, 1, Y, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
SQ11 2010-Jan-19, SIN-BKK 2010-Jan-19, Y, 2, M, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
*******************************************
*******************************************
FlightDate: SQ11, 2010-Jan-20
*******************************************
*******************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
      apsed, Distance, Capacity,
SQ11 2010-Jan-20, SIN-BKK, 2010-Jan-20, 08:20:00, 2010-Jan-20, 11:00:00, 07:40:00
      , 0, -05:00:00, 6300, 0,
*******************************************
*******************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
      Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ11 2010-Jan-20, SIN-BKK 2010-Jan-20, Y, 300, 300, 0, 0, 0, 0, 0, 0, 0, 0, 300,
      9, 0, 0, 0, 0, 0,
*******************************************
*******************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ11 2010-Jan-20, SIN-BKK 2010-Jan-20, Y, 1, 0, 0, 0, 0, 300, 0,
SQ11 2010-Jan-20, SIN-BKK 2010-Jan-20, Y, 2, 0, 0, 0, 0, 300, 0,
*******************************************
*******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
      (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ11 2010-Jan-20, SIN-BKK 2010-Jan-20, Y, 1, Y, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
```

```
       0, 0, 0, 0,
SQ11 2010-Jan-20, SIN-BKK 2010-Jan-20, Y, 2, M, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
       0, 0, 0, 0,
*******************************************
*******************************************
FlightDate: SQ11, 2010-Jan-21
*******************************************
*******************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
       apsed, Distance, Capacity,
SQ11 2010-Jan-21, SIN-BKK, 2010-Jan-21, 08:20:00, 2010-Jan-21, 11:00:00, 07:40:00
       , 0, -05:00:00, 6300, 0,
*******************************************
*******************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
       Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ11 2010-Jan-21, SIN-BKK 2010-Jan-21, Y, 300, 300, 0, 0, 0, 0, 0, 0, 0, 0, 300,
       9, 0, 0, 0, 0, 0,
*******************************************
*******************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ11 2010-Jan-21, SIN-BKK 2010-Jan-21, Y, 1, 0, 0, 0, 0, 300, 0,
SQ11 2010-Jan-21, SIN-BKK 2010-Jan-21, Y, 2, 0, 0, 0, 0, 300, 0,
*******************************************
*******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
       (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ11 2010-Jan-21, SIN-BKK 2010-Jan-21, Y, 1, Y, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
       0, 0, 0, 0,
SQ11 2010-Jan-21, SIN-BKK 2010-Jan-21, Y, 2, M, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
       0, 0, 0, 0,
*******************************************
*******************************************
FlightDate: SQ11, 2010-Jan-22
*******************************************
*******************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
       apsed, Distance, Capacity,
SQ11 2010-Jan-22, SIN-BKK, 2010-Jan-22, 08:20:00, 2010-Jan-22, 11:00:00, 07:40:00
       , 0, -05:00:00, 6300, 0,
*******************************************
*******************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
       Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ11 2010-Jan-22, SIN-BKK 2010-Jan-22, Y, 300, 300, 0, 0, 0, 0, 0, 0, 0, 0, 300,
```

```
       9, 0, 0, 0, 0, 0,
*******************************************
*******************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ11 2010-Jan-22, SIN-BKK 2010-Jan-22, Y, 1, 0, 0, 0, 0, 300, 0,
SQ11 2010-Jan-22, SIN-BKK 2010-Jan-22, Y, 2, 0, 0, 0, 0, 300, 0,
*******************************************
*******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
       (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ11 2010-Jan-22, SIN-BKK 2010-Jan-22, Y, 1, Y, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
       0, 0, 0, 0,
SQ11 2010-Jan-22, SIN-BKK 2010-Jan-22, Y, 2, M, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
       0, 0, 0, 0,
*******************************************
*******************************************
FlightDate: SQ11, 2010-Jan-23
*******************************************
*******************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
       apsed, Distance, Capacity,
SQ11 2010-Jan-23, SIN-BKK, 2010-Jan-23, 08:20:00, 2010-Jan-23, 11:00:00, 07:40:00
       , 0, -05:00:00, 6300, 0,
*******************************************
*******************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
       Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ11 2010-Jan-23, SIN-BKK 2010-Jan-23, Y, 300, 300, 0, 0, 0, 0, 0, 0, 6.64029e-31
       9, 0, 300, 9, 0, 0, 0, 0, 0,
*******************************************
*******************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ11 2010-Jan-23, SIN-BKK 2010-Jan-23, Y, 1, 0, 0, 0, 0, 300, 0,
SQ11 2010-Jan-23, SIN-BKK 2010-Jan-23, Y, 2, 0, 0, 0, 0, 300, 0,
*******************************************
*******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
       (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ11 2010-Jan-23, SIN-BKK 2010-Jan-23, Y, 1, Y, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
       0, 0, 0, 0,
```

```
SQ11 2010-Jan-23, SIN-BKK 2010-Jan-23, Y, 2, M, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
*******************************************
*******************************************
FlightDate: SQ11, 2010-Jan-24
*******************************************
*******************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
      apsed, Distance, Capacity,
SQ11 2010-Jan-24, SIN-BKK, 2010-Jan-24, 08:20:00, 2010-Jan-24, 11:00:00, 07:40:00
      , 0, -05:00:00, 6300, 0,
*******************************************
*******************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
      Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ11 2010-Jan-24, SIN-BKK 2010-Jan-24, Y, 300, 300, 0, 0, 0, 0, 0, 0, 0, 0, 300,
      9, 0, 0, 0, 0, 0,
*******************************************
*******************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ11 2010-Jan-24, SIN-BKK 2010-Jan-24, Y, 1, 0, 0, 0, 0, 300, 0,
SQ11 2010-Jan-24, SIN-BKK 2010-Jan-24, Y, 2, 0, 0, 0, 0, 300, 0,
*******************************************
*******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
      (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ11 2010-Jan-24, SIN-BKK 2010-Jan-24, Y, 1, Y, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
SQ11 2010-Jan-24, SIN-BKK 2010-Jan-24, Y, 2, M, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
*******************************************
*******************************************
FlightDate: SQ11, 2010-Jan-25
*******************************************
*******************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
      apsed, Distance, Capacity,
SQ11 2010-Jan-25, SIN-BKK, 2010-Jan-25, 08:20:00, 2010-Jan-25, 11:00:00, 07:40:00
      , 0, -05:00:00, 6300, 0,
*******************************************
*******************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
      Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ11 2010-Jan-25, SIN-BKK 2010-Jan-25, Y, 300, 300, 0, 0, 0, 0, 0, 0, 0, 0, 300,
      9, 0, 0, 0, 0, 0,
```

```
*******************************************
*******************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ11 2010-Jan-25, SIN-BKK 2010-Jan-25, Y, 1, 0, 0, 0, 0, 300, 0,
SQ11 2010-Jan-25, SIN-BKK 2010-Jan-25, Y, 2, 0, 0, 0, 0, 300, 0,
*******************************************
*******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
      (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ11 2010-Jan-25, SIN-BKK 2010-Jan-25, Y, 1, Y, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
SQ11 2010-Jan-25, SIN-BKK 2010-Jan-25, Y, 2, M, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
*******************************************
*******************************************
FlightDate: SQ11, 2010-Jan-26
*******************************************
*******************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
      apsed, Distance, Capacity,
SQ11 2010-Jan-26, SIN-BKK, 2010-Jan-26, 08:20:00, 2010-Jan-26, 11:00:00, 07:40:00
      , 0, -05:00:00, 6300, 0,
*******************************************
*******************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
      Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ11 2010-Jan-26, SIN-BKK 2010-Jan-26, Y, 300, 300, 0, 0, 0, 0, 0, 0, 0, 0, 300,
      9, 0, 0, 0, 0, 0,
*******************************************
*******************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ11 2010-Jan-26, SIN-BKK 2010-Jan-26, Y, 1, 0, 0, 0, 0, 300, 0,
SQ11 2010-Jan-26, SIN-BKK 2010-Jan-26, Y, 2, 0, 0, 0, 0, 300, 0,
*******************************************
*******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
      (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ11 2010-Jan-26, SIN-BKK 2010-Jan-26, Y, 1, Y, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
SQ11 2010-Jan-26, SIN-BKK 2010-Jan-26, Y, 2, M, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
```

```
      0, 0, 0, 0,
*******************************************
*******************************************
FlightDate: SQ11, 2010-Jan-27
*******************************************
*******************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
      apsed, Distance, Capacity,
SQ11 2010-Jan-27, SIN-BKK, 2010-Jan-27, 08:20:00, 2010-Jan-27, 11:00:00, 07:40:00
      , 0, -05:00:00, 6300, 0,
*******************************************
*******************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
      Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ11 2010-Jan-27, SIN-BKK 2010-Jan-27, Y, 300, 300, 0, 0, 0, 0, 0, 0, 0, 0, 300,
      9, 0, 0, 0, 0, 0,
*******************************************
*******************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ11 2010-Jan-27, SIN-BKK 2010-Jan-27, Y, 1, 0, 0, 0, 0, 300, 0,
SQ11 2010-Jan-27, SIN-BKK 2010-Jan-27, Y, 2, 0, 0, 0, 0, 300, 0,
*******************************************
*******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
      (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ11 2010-Jan-27, SIN-BKK 2010-Jan-27, Y, 1, Y, 300 (0), 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
SQ11 2010-Jan-27, SIN-BKK 2010-Jan-27, Y, 2, M, 300 (0), 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
*******************************************
*******************************************
FlightDate: SQ11, 2010-Jan-28
*******************************************
*******************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
      apsed, Distance, Capacity,
SQ11 2010-Jan-28, SIN-BKK, 2010-Jan-28, 08:20:00, 2010-Jan-28, 11:00:00, 07:40:00
      , 0, -05:00:00, 6300, 0,
*******************************************
*******************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
      Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ11 2010-Jan-28, SIN-BKK 2010-Jan-28, Y, 300, 300, 0, 0, 0, 0, 0, 0, 0, 0, 300,
      9, 0, 0, 0, 0, 0,
*******************************************
```

```
*******************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ11 2010-Jan-28, SIN-BKK 2010-Jan-28, Y, 1, 0, 0, 0, 0, 300, 0,
SQ11 2010-Jan-28, SIN-BKK 2010-Jan-28, Y, 2, 0, 0, 0, 0, 300, 0,
*******************************************
*******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
      (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ11 2010-Jan-28, SIN-BKK 2010-Jan-28, Y, 1, Y, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
SQ11 2010-Jan-28, SIN-BKK 2010-Jan-28, Y, 2, M, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
*******************************************
*******************************************
FlightDate: SQ11, 2010-Jan-29
*******************************************
*******************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
      apsed, Distance, Capacity,
SQ11 2010-Jan-29, SIN-BKK, 2010-Jan-29, 08:20:00, 2010-Jan-29, 11:00:00, 07:40:00
      , 0, -05:00:00, 6300, 0,
*******************************************
*******************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
      Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ11 2010-Jan-29, SIN-BKK 2010-Jan-29, Y, 300, 300, 0, 0, 0, 0, 0, 0, 0, 0, 300,
      9, 0, 0, 0, 0, 0,
*******************************************
*******************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ11 2010-Jan-29, SIN-BKK 2010-Jan-29, Y, 1, 0, 0, 0, 0, 300, 0,
SQ11 2010-Jan-29, SIN-BKK 2010-Jan-29, Y, 2, 0, 0, 0, 0, 300, 0,
*******************************************
*******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
      (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ11 2010-Jan-29, SIN-BKK 2010-Jan-29, Y, 1, Y, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
SQ11 2010-Jan-29, SIN-BKK 2010-Jan-29, Y, 2, M, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
```

```
*******************************************
*******************************************
FlightDate: SQ11, 2010-Jan-30
*******************************************
*******************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
      apsed, Distance, Capacity,
SQ11 2010-Jan-30, SIN-BKK, 2010-Jan-30, 08:20:00, 2010-Jan-30, 11:00:00, 07:40:00
      , 0, -05:00:00, 6300, 0,
*******************************************
*******************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
      Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ11 2010-Jan-30, SIN-BKK 2010-Jan-30, Y, 300, 300, 0, 0, 0, 0, 0, 0, 0, 0, 300,
      9, 0, 0, 0, 0, 0,
*******************************************
*******************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ11 2010-Jan-30, SIN-BKK 2010-Jan-30, Y, 1, 0, 0, 0, 0, 300, 0,
SQ11 2010-Jan-30, SIN-BKK 2010-Jan-30, Y, 2, 0, 0, 0, 0, 300, 0,
*******************************************
*******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
      (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ11 2010-Jan-30, SIN-BKK 2010-Jan-30, Y, 1, Y, 300 (0), 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
SQ11 2010-Jan-30, SIN-BKK 2010-Jan-30, Y, 2, M, 300 (0), 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
*******************************************
*******************************************
FlightDate: SQ11, 2010-Jan-31
*******************************************
*******************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
      apsed, Distance, Capacity,
SQ11 2010-Jan-31, SIN-BKK, 2010-Jan-31, 08:20:00, 2010-Jan-31, 11:00:00, 07:40:00
      , 0, -05:00:00, 6300, 0,
*******************************************
*******************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
      Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ11 2010-Jan-31, SIN-BKK 2010-Jan-31, Y, 300, 300, 0, 0, 0, 0, 0, 0, 0, 0, 300,
      9, 0, 0, 0, 0, 0,
*******************************************
*******************************************
```

```
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ11 2010-Jan-31, SIN-BKK 2010-Jan-31, Y, 1, 0, 0, 0, 0, 300, 0,
SQ11 2010-Jan-31, SIN-BKK 2010-Jan-31, Y, 2, 0, 0, 0, 0, 300, 0,
*******************************************
*******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
       (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ11 2010-Jan-31, SIN-BKK 2010-Jan-31, Y, 1, Y, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
       0, 0, 0, 0,
SQ11 2010-Jan-31, SIN-BKK 2010-Jan-31, Y, 2, M, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
       0, 0, 0, 0,
*******************************************
*******************************************
FlightDate: SQ11, 2010-Feb-01
*******************************************
*******************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
      apsed, Distance, Capacity,
SQ11 2010-Feb-01, SIN-BKK, 2010-Feb-01, 08:20:00, 2010-Feb-01, 11:00:00, 07:40:00
      , 0, -05:00:00, 6300, 0,
*******************************************
*******************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
      Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ11 2010-Feb-01, SIN-BKK 2010-Feb-01, Y, 300, 300, 0, 0, 0, 0, 0, 0, 0, 300,
      9, 0, 0, 0, 0, 0,
*******************************************
*******************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ11 2010-Feb-01, SIN-BKK 2010-Feb-01, Y, 1, 0, 0, 0, 0, 300, 0,
SQ11 2010-Feb-01, SIN-BKK 2010-Feb-01, Y, 2, 0, 0, 0, 0, 300, 0,
*******************************************
*******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
       (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ11 2010-Feb-01, SIN-BKK 2010-Feb-01, Y, 1, Y, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
       0, 0, 0, 0,
SQ11 2010-Feb-01, SIN-BKK 2010-Feb-01, Y, 2, M, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
       0, 0, 0, 0,
*******************************************
```

```
*******************************************
FlightDate: SQ11, 2010-Feb-02
*******************************************
*******************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
      apsed, Distance, Capacity,
SQ11 2010-Feb-02, SIN-BKK, 2010-Feb-02, 08:20:00, 2010-Feb-02, 11:00:00, 07:40:00
      , 0, -05:00:00, 6300, 0,
*******************************************
*******************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
      Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ11 2010-Feb-02, SIN-BKK 2010-Feb-02, Y, 300, 300, 0, 0, 0, 0, 0, 0, 0, 0, 300,
      9, 0, 0, 0, 0, 0,
*******************************************
*******************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ11 2010-Feb-02, SIN-BKK 2010-Feb-02, Y, 1, 0, 0, 0, 0, 300, 0,
SQ11 2010-Feb-02, SIN-BKK 2010-Feb-02, Y, 2, 0, 0, 0, 0, 300, 0,
*******************************************
*******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
        (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ11 2010-Feb-02, SIN-BKK 2010-Feb-02, Y, 1, Y, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
SQ11 2010-Feb-02, SIN-BKK 2010-Feb-02, Y, 2, M, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
*******************************************
*******************************************
FlightDate: SQ11, 2010-Feb-03
*******************************************
*******************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
      apsed, Distance, Capacity,
SQ11 2010-Feb-03, SIN-BKK, 2010-Feb-03, 08:20:00, 2010-Feb-03, 11:00:00, 07:40:00
      , 0, -05:00:00, 6300, 0,
*******************************************
*******************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
      Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ11 2010-Feb-03, SIN-BKK 2010-Feb-03, Y, 300, 300, 0, 0, 0, 0, 0, 0, 0, 0, 300,
      9, 0, 0, 0, 0, 0,
*******************************************
*******************************************
Buckets:
```

```
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ11 2010-Feb-03, SIN-BKK 2010-Feb-03, Y, 1, 0, 0, 0, 0, 300, 0,
SQ11 2010-Feb-03, SIN-BKK 2010-Feb-03, Y, 2, 0, 0, 0, 0, 300, 0,
*******************************************
*******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
       (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ11 2010-Feb-03, SIN-BKK 2010-Feb-03, Y, 1, Y, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
       0, 0, 0, 0,
SQ11 2010-Feb-03, SIN-BKK 2010-Feb-03, Y, 2, M, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
       0, 0, 0, 0,
*******************************************
*******************************************
FlightDate: SQ11, 2010-Feb-04
*******************************************
*******************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
       apsed, Distance, Capacity,
SQ11 2010-Feb-04, SIN-BKK, 2010-Feb-04, 08:20:00, 2010-Feb-04, 11:00:00, 07:40:00
       , 0, -05:00:00, 6300, 0,
*******************************************
*******************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
       Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ11 2010-Feb-04, SIN-BKK 2010-Feb-04, Y, 300, 300, 0, 0, 0, 0, 0, 0, 0, 0, 300,
       9, 0, 0, 0, 0, 0,
*******************************************
*******************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ11 2010-Feb-04, SIN-BKK 2010-Feb-04, Y, 1, 0, 0, 0, 0, 300, 0,
SQ11 2010-Feb-04, SIN-BKK 2010-Feb-04, Y, 2, 0, 0, 0, 0, 300, 0,
*******************************************
*******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
       (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ11 2010-Feb-04, SIN-BKK 2010-Feb-04, Y, 1, Y, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
       0, 0, 0, 0,
SQ11 2010-Feb-04, SIN-BKK 2010-Feb-04, Y, 2, M, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
       0, 0, 0, 0,
*******************************************
*******************************************
```

```
FlightDate: SQ11, 2010-Feb-05
*******************************************
*******************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
      apsed, Distance, Capacity,
SQ11 2010-Feb-05, SIN-BKK, 2010-Feb-05, 08:20:00, 2010-Feb-05, 11:00:00, 07:40:00
      , 0, -05:00:00, 6300, 0,
*******************************************
*******************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
      Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ11 2010-Feb-05, SIN-BKK 2010-Feb-05, Y, 300, 300, 0, 0, 0, 0, 0, 0, 0, 0, 300,
      9, 0, 0, 0, 0, 0,
*******************************************
*******************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ11 2010-Feb-05, SIN-BKK 2010-Feb-05, Y, 1, 0, 0, 0, 0, 300, 0,
SQ11 2010-Feb-05, SIN-BKK 2010-Feb-05, Y, 2, 0, 0, 0, 0, 300, 0,
*******************************************
*******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
      (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ11 2010-Feb-05, SIN-BKK 2010-Feb-05, Y, 1, Y, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
SQ11 2010-Feb-05, SIN-BKK 2010-Feb-05, Y, 2, M, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
*******************************************
*******************************************
FlightDate: SQ11, 2010-Feb-06
*******************************************
*******************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
      apsed, Distance, Capacity,
SQ11 2010-Feb-06, SIN-BKK, 2010-Feb-06, 08:20:00, 2010-Feb-06, 11:00:00, 07:40:00
      , 0, -05:00:00, 6300, 0,
*******************************************
*******************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
      Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ11 2010-Feb-06, SIN-BKK 2010-Feb-06, Y, 300, 300, 0, 0, 0, 0, 0, 0, 0, 0, 300,
      9, 0, 0, 0, 0, 0,
*******************************************
*******************************************
Buckets:
--------
```

```
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ11 2010-Feb-06, SIN-BKK 2010-Feb-06, Y, 1, 0, 0, 0, 0, 300, 0,
SQ11 2010-Feb-06, SIN-BKK 2010-Feb-06, Y, 2, 0, 0, 0, 0, 300, 0,
*******************************************
*******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
        (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ11 2010-Feb-06, SIN-BKK 2010-Feb-06, Y, 1, Y, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
        0, 0, 0, 0,
SQ11 2010-Feb-06, SIN-BKK 2010-Feb-06, Y, 2, M, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
        0, 0, 0, 0,
*******************************************
*******************************************
FlightDate: SQ11, 2010-Feb-07
*******************************************
*******************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
        apsed, Distance, Capacity,
SQ11 2010-Feb-07, SIN-BKK, 2010-Feb-07, 08:20:00, 2010-Feb-07, 11:00:00, 07:40:00
        , 0, -05:00:00, 6300, 0,
*******************************************
*******************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
        Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ11 2010-Feb-07, SIN-BKK 2010-Feb-07, Y, 300, 300, 0, 0, 0, 0, 0, 0, 0, 0, 300,
        9, 0, 0, 0, 0, 0,
*******************************************
*******************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ11 2010-Feb-07, SIN-BKK 2010-Feb-07, Y, 1, 0, 0, 0, 0, 300, 0,
SQ11 2010-Feb-07, SIN-BKK 2010-Feb-07, Y, 2, 0, 0, 0, 0, 300, 0,
*******************************************
*******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
        (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ11 2010-Feb-07, SIN-BKK 2010-Feb-07, Y, 1, Y, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
        0, 0, 0, 0,
SQ11 2010-Feb-07, SIN-BKK 2010-Feb-07, Y, 2, M, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
        0, 0, 0, 0,
*******************************************
*******************************************
FlightDate: SQ11, 2010-Feb-08
```

```
*******************************************
*******************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
      apsed, Distance, Capacity,
SQ11 2010-Feb-08, SIN-BKK, 2010-Feb-08, 08:20:00, 2010-Feb-08, 11:00:00, 07:40:00
      , 0, -05:00:00, 6300, 0,
*******************************************
*******************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
      Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ11 2010-Feb-08, SIN-BKK 2010-Feb-08, Y, 300, 300, 0, 0, 0, 0, 0, 0, 0, 0, 300,
      9, 0, 0, 0, 0, 0,
*******************************************
*******************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ11 2010-Feb-08, SIN-BKK 2010-Feb-08, Y, 1, 0, 0, 0, 0, 300, 0,
SQ11 2010-Feb-08, SIN-BKK 2010-Feb-08, Y, 2, 0, 0, 0, 0, 300, 0,
*******************************************
*******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
      (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ11 2010-Feb-08, SIN-BKK 2010-Feb-08, Y, 1, Y, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
SQ11 2010-Feb-08, SIN-BKK 2010-Feb-08, Y, 2, M, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
*******************************************
*******************************************
FlightDate: SQ11, 2010-Feb-09
*******************************************
*******************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
      apsed, Distance, Capacity,
SQ11 2010-Feb-09, SIN-BKK, 2010-Feb-09, 08:20:00, 2010-Feb-09, 11:00:00, 07:40:00
      , 0, -05:00:00, 6300, 0,
*******************************************
*******************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
      Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ11 2010-Feb-09, SIN-BKK 2010-Feb-09, Y, 300, 300, 0, 0, 0, 0, 0, 0, 0, 0, 300,
      9, 0, 0, 0, 0, 0,
*******************************************
*******************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
```

```
*******************************************
*******************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ11 2010-Feb-09, SIN-BKK 2010-Feb-09, Y, 1, 0, 0, 0, 0, 300, 0,
SQ11 2010-Feb-09, SIN-BKK 2010-Feb-09, Y, 2, 0, 0, 0, 0, 300, 0,
*******************************************
*******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
        (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ11 2010-Feb-09, SIN-BKK 2010-Feb-09, Y, 1, Y, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
        0, 0, 0, 0,
SQ11 2010-Feb-09, SIN-BKK 2010-Feb-09, Y, 2, M, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
        0, 0, 0, 0,
*******************************************
*******************************************
FlightDate: SQ11, 2010-Feb-10
*******************************************
*******************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
        apsed, Distance, Capacity,
SQ11 2010-Feb-10, SIN-BKK, 2010-Feb-10, 08:20:00, 2010-Feb-10, 11:00:00, 07:40:00
        , 0, -05:00:00, 6300, 0,
*******************************************
*******************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
        Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ11 2010-Feb-10, SIN-BKK 2010-Feb-10, Y, 300, 300, 0, 0, 0, 0, 0, 0, 0, 0, 300,
        9, 0, 0, 0, 0, 0,
*******************************************
*******************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ11 2010-Feb-10, SIN-BKK 2010-Feb-10, Y, 1, 0, 0, 0, 0, 300, 0,
SQ11 2010-Feb-10, SIN-BKK 2010-Feb-10, Y, 2, 0, 0, 0, 0, 300, 0,
*******************************************
*******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
        (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ11 2010-Feb-10, SIN-BKK 2010-Feb-10, Y, 1, Y, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
        0, 0, 0, 0,
SQ11 2010-Feb-10, SIN-BKK 2010-Feb-10, Y, 2, M, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
        0, 0, 0, 0,
*******************************************
*******************************************
FlightDate: SQ11, 2010-Feb-11
*******************************************
```

```
*******************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
      apsed, Distance, Capacity,
SQ11 2010-Feb-11, SIN-BKK, 2010-Feb-11, 08:20:00, 2010-Feb-11, 11:00:00, 07:40:00
      , 0, -05:00:00, 6300, 0,
*******************************************
*******************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
      Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ11 2010-Feb-11, SIN-BKK 2010-Feb-11, Y, 300, 300, 0, 0, 0, 0, 0, 0, 0, 0, 300,
      9, 0, 0, 0, 0, 0,
*******************************************
*******************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ11 2010-Feb-11, SIN-BKK 2010-Feb-11, Y, 1, 0, 0, 0, 0, 300, 0,
SQ11 2010-Feb-11, SIN-BKK 2010-Feb-11, Y, 2, 0, 0, 0, 0, 300, 0,
*******************************************
*******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
       (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ11 2010-Feb-11, SIN-BKK 2010-Feb-11, Y, 1, Y, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
SQ11 2010-Feb-11, SIN-BKK 2010-Feb-11, Y, 2, M, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
*******************************************
*******************************************
FlightDate: SQ11, 2010-Feb-12
*******************************************
*******************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
      apsed, Distance, Capacity,
SQ11 2010-Feb-12, SIN-BKK, 2010-Feb-12, 08:20:00, 2010-Feb-12, 11:00:00, 07:40:00
      , 0, -05:00:00, 6300, 0,
*******************************************
*******************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
      Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ11 2010-Feb-12, SIN-BKK 2010-Feb-12, Y, 300, 300, 0, 0, 0, 0, 0, 0, 0, 0, 300,
      9, 0, 0, 0, 0, 0,
*******************************************
*******************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
```

```
*******************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ11 2010-Feb-12, SIN-BKK 2010-Feb-12, Y, 1, 0, 0, 0, 0, 300, 0,
SQ11 2010-Feb-12, SIN-BKK 2010-Feb-12, Y, 2, 0, 0, 0, 0, 300, 0,
*******************************************
*******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
       (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ11 2010-Feb-12, SIN-BKK 2010-Feb-12, Y, 1, Y, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
       0, 0, 0, 0,
SQ11 2010-Feb-12, SIN-BKK 2010-Feb-12, Y, 2, M, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
       0, 0, 0, 0,
*******************************************
*******************************************
FlightDate: SQ11, 2010-Feb-13
*******************************************
*******************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
       apsed, Distance, Capacity,
SQ11 2010-Feb-13, SIN-BKK, 2010-Feb-13, 08:20:00, 2010-Feb-13, 11:00:00, 07:40:00
       , 0, -05:00:00, 6300, 0,
*******************************************
*******************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
       Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ11 2010-Feb-13, SIN-BKK 2010-Feb-13, Y, 300, 300, 0, 0, 0, 0, 0, 0, 0, 0, 300,
       9, 0, 0, 0, 0, 0,
*******************************************
*******************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ11 2010-Feb-13, SIN-BKK 2010-Feb-13, Y, 1, 0, 0, 0, 0, 300, 0,
SQ11 2010-Feb-13, SIN-BKK 2010-Feb-13, Y, 2, 0, 0, 0, 0, 300, 0,
*******************************************
*******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
       (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ11 2010-Feb-13, SIN-BKK 2010-Feb-13, Y, 1, Y, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
       0, 0, 0, 0,
SQ11 2010-Feb-13, SIN-BKK 2010-Feb-13, Y, 2, M, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
       0, 0, 0, 0,
*******************************************
*******************************************
FlightDate: SQ11, 2010-Feb-14
*******************************************
*******************************************
```

```
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
     apsed, Distance, Capacity,
SQ11 2010-Feb-14, SIN-BKK, 2010-Feb-14, 08:20:00, 2010-Feb-14, 11:00:00, 07:40:00
     , 0, -05:00:00, 6300, 0,
*******************************************
*******************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
     Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ11 2010-Feb-14, SIN-BKK 2010-Feb-14, Y, 300, 300, 0, 0, 0, 0, 0, 0, 0, 0, 300,
     9, 0, 0, 0, 0, 0,
*******************************************
*******************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ11 2010-Feb-14, SIN-BKK 2010-Feb-14, Y, 1, 0, 0, 0, 0, 300, 0,
SQ11 2010-Feb-14, SIN-BKK 2010-Feb-14, Y, 2, 0, 0, 0, 0, 300, 0,
*******************************************
*******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
      (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ11 2010-Feb-14, SIN-BKK 2010-Feb-14, Y, 1, Y, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
     0, 0, 0, 0,
SQ11 2010-Feb-14, SIN-BKK 2010-Feb-14, Y, 2, M, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
     0, 0, 0, 0,
*******************************************
*******************************************
FlightDate: SQ11, 2010-Feb-15
*******************************************
*******************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
     apsed, Distance, Capacity,
SQ11 2010-Feb-15, SIN-BKK, 2010-Feb-15, 08:20:00, 2010-Feb-15, 11:00:00, 07:40:00
     , 0, -05:00:00, 6300, 0,
*******************************************
*******************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
     Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ11 2010-Feb-15, SIN-BKK 2010-Feb-15, Y, 300, 300, 0, 0, 0, 0, 0, 0, 0, 0, 300,
     9, 0, 0, 0, 0, 0,
*******************************************
*******************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
```

```
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ11 2010-Feb-15, SIN-BKK 2010-Feb-15, Y, 1, 0, 0, 0, 0, 300, 0,
SQ11 2010-Feb-15, SIN-BKK 2010-Feb-15, Y, 2, 0, 0, 0, 0, 300, 0,
******************************************
******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
       (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ11 2010-Feb-15, SIN-BKK 2010-Feb-15, Y, 1, Y, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
       0, 0, 0, 0,
SQ11 2010-Feb-15, SIN-BKK 2010-Feb-15, Y, 2, M, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
       0, 0, 0, 0,
******************************************
******************************************
FlightDate: SQ11, 2010-Feb-16
******************************************
******************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
       apsed, Distance, Capacity,
SQ11 2010-Feb-16, SIN-BKK, 2010-Feb-16, 08:20:00, 2010-Feb-16, 11:00:00, 07:40:00
       , 0, -05:00:00, 6300, 0,
******************************************
******************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
       Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ11 2010-Feb-16, SIN-BKK 2010-Feb-16, Y, 300, 300, 0, 0, 0, 0, 0, 0, 0, 0, 300,
       9, 0, 0, 0, 0, 0,
******************************************
******************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
******************************************
******************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ11 2010-Feb-16, SIN-BKK 2010-Feb-16, Y, 1, 0, 0, 0, 0, 300, 0,
SQ11 2010-Feb-16, SIN-BKK 2010-Feb-16, Y, 2, 0, 0, 0, 0, 300, 0,
******************************************
******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
       (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ11 2010-Feb-16, SIN-BKK 2010-Feb-16, Y, 1, Y, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
       0, 0, 0, 0,
SQ11 2010-Feb-16, SIN-BKK 2010-Feb-16, Y, 2, M, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
       0, 0, 0, 0,
******************************************
******************************************
FlightDate: SQ11, 2010-Feb-17
******************************************
******************************************
Leg-Dates:
```

```
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
     apsed, Distance, Capacity,
SQ11 2010-Feb-17, SIN-BKK, 2010-Feb-17, 08:20:00, 2010-Feb-17, 11:00:00, 07:40:00
     , 0, -05:00:00, 6300, 0,
*****************************************
*****************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
     Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ11 2010-Feb-17, SIN-BKK 2010-Feb-17, Y, 300, 300, 0, 0, 0, 0, 0, 0, 0, 0, 300,
     9, 0, 0, 0, 0, 0,
*****************************************
*****************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*****************************************
*****************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ11 2010-Feb-17, SIN-BKK 2010-Feb-17, Y, 1, 0, 0, 0, 0, 300, 0,
SQ11 2010-Feb-17, SIN-BKK 2010-Feb-17, Y, 2, 0, 0, 0, 0, 300, 0,
*****************************************
*****************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
      (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ11 2010-Feb-17, SIN-BKK 2010-Feb-17, Y, 1, Y, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
     0, 0, 0, 0,
SQ11 2010-Feb-17, SIN-BKK 2010-Feb-17, Y, 2, M, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
     0, 0, 0, 0,
*****************************************
*****************************************
FlightDate: SQ11, 2010-Feb-18
*****************************************
*****************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
     apsed, Distance, Capacity,
SQ11 2010-Feb-18, SIN-BKK, 2010-Feb-18, 08:20:00, 2010-Feb-18, 11:00:00, 07:40:00
     , 0, -05:00:00, 6300, 0,
*****************************************
*****************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
     Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ11 2010-Feb-18, SIN-BKK 2010-Feb-18, Y, 300, 300, 0, 0, 0, 0, 0, 0, 0, 0, 300,
     9, 0, 0, 0, 0, 0,
*****************************************
*****************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*****************************************
*****************************************
SegmentCabins:
```

```
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ11 2010-Feb-18, SIN-BKK 2010-Feb-18, Y, 1, 0, 0, 0, 0, 300, 0,
SQ11 2010-Feb-18, SIN-BKK 2010-Feb-18, Y, 2, 0, 0, 0, 0, 300, 0,
*******************************************
*******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
       (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ11 2010-Feb-18, SIN-BKK 2010-Feb-18, Y, 1, Y, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
       0, 0, 0, 0,
SQ11 2010-Feb-18, SIN-BKK 2010-Feb-18, Y, 2, M, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
       0, 0, 0, 0,
*******************************************
*******************************************
FlightDate: SQ11, 2010-Feb-19
*******************************************
*******************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
      apsed, Distance, Capacity,
SQ11 2010-Feb-19, SIN-BKK, 2010-Feb-19, 08:20:00, 2010-Feb-19, 11:00:00, 07:40:00
       , 0, -05:00:00, 6300, 0,
*******************************************
*******************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
      Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ11 2010-Feb-19, SIN-BKK 2010-Feb-19, Y, 300, 300, 0, 0, 0, 0, 0, 0, 0, 300,
       9, 0, 0, 0, 0, 0,
*******************************************
*******************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ11 2010-Feb-19, SIN-BKK 2010-Feb-19, Y, 1, 0, 0, 0, 0, 300, 0,
SQ11 2010-Feb-19, SIN-BKK 2010-Feb-19, Y, 2, 0, 0, 0, 0, 300, 0,
*******************************************
*******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
       (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ11 2010-Feb-19, SIN-BKK 2010-Feb-19, Y, 1, Y, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
       0, 0, 0, 0,
SQ11 2010-Feb-19, SIN-BKK 2010-Feb-19, Y, 2, M, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
       0, 0, 0, 0,
*******************************************
*******************************************
FlightDate: SQ11, 2010-Feb-20
*******************************************
*******************************************
Leg-Dates:
----------
```

```
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
      apsed, Distance, Capacity,
SQ11 2010-Feb-20, SIN-BKK, 2010-Feb-20, 08:20:00, 2010-Feb-20, 11:00:00, 07:40:00
      , 0, -05:00:00, 6300, 0,
*******************************************
*******************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
      Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ11 2010-Feb-20, SIN-BKK 2010-Feb-20, Y, 300, 300, 0, 0, 0, 0, 0, 0, 0, 0, 300,
      9, 0, 0, 0, 0, 0,
*******************************************
*******************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ11 2010-Feb-20, SIN-BKK 2010-Feb-20, Y, 1, 0, 0, 0, 0, 300, 0,
SQ11 2010-Feb-20, SIN-BKK 2010-Feb-20, Y, 2, 0, 0, 0, 0, 300, 0,
*******************************************
*******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
      (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ11 2010-Feb-20, SIN-BKK 2010-Feb-20, Y, 1, Y, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
SQ11 2010-Feb-20, SIN-BKK 2010-Feb-20, Y, 2, M, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
*******************************************
*******************************************
FlightDate: SQ11, 2010-Feb-21
*******************************************
*******************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
      apsed, Distance, Capacity,
SQ11 2010-Feb-21, SIN-BKK, 2010-Feb-21, 08:20:00, 2010-Feb-21, 11:00:00, 07:40:00
      , 0, -05:00:00, 6300, 0,
*******************************************
*******************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
      Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ11 2010-Feb-21, SIN-BKK 2010-Feb-21, Y, 300, 300, 0, 0, 0, 0, 0, 0, 0, 0, 300,
      9, 0, 0, 0, 0, 0,
*******************************************
*******************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
SegmentCabins:
--------------
```

```
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ11 2010-Feb-21, SIN-BKK 2010-Feb-21, Y, 1, 0, 0, 0, 0, 300, 0,
SQ11 2010-Feb-21, SIN-BKK 2010-Feb-21, Y, 2, 0, 0, 0, 0, 300, 0,
*******************************************
*******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
        (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ11 2010-Feb-21, SIN-BKK 2010-Feb-21, Y, 1, Y, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
        0, 0, 0, 0,
SQ11 2010-Feb-21, SIN-BKK 2010-Feb-21, Y, 2, M, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
        0, 0, 0, 0,
*******************************************
*******************************************
FlightDate: SQ11, 2010-Feb-22
*******************************************
*******************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
        apsed, Distance, Capacity,
SQ11 2010-Feb-22, SIN-BKK, 2010-Feb-22, 08:20:00, 2010-Feb-22, 11:00:00, 07:40:00
        , 0, -05:00:00, 6300, 0,
*******************************************
*******************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
        Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ11 2010-Feb-22, SIN-BKK 2010-Feb-22, Y, 300, 300, 0, 0, 0, 0, 0, 0, 0, 0, 300,
        9, 0, 0, 0, 0, 0,
*******************************************
*******************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ11 2010-Feb-22, SIN-BKK 2010-Feb-22, Y, 1, 0, 0, 0, 0, 300, 0,
SQ11 2010-Feb-22, SIN-BKK 2010-Feb-22, Y, 2, 0, 0, 0, 0, 300, 0,
*******************************************
*******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
        (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ11 2010-Feb-22, SIN-BKK 2010-Feb-22, Y, 1, Y, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
        0, 0, 0, 0,
SQ11 2010-Feb-22, SIN-BKK 2010-Feb-22, Y, 2, M, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
        0, 0, 0, 0,
*******************************************
*******************************************
FlightDate: SQ11, 2010-Feb-23
*******************************************
*******************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
```

```
        apsed, Distance, Capacity,
SQ11 2010-Feb-23, SIN-BKK, 2010-Feb-23, 08:20:00, 2010-Feb-23, 11:00:00, 07:40:00
        , 0, -05:00:00, 6300, 0,
********************************************
********************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
        Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ11 2010-Feb-23, SIN-BKK 2010-Feb-23, Y, 300, 300, 0, 0, 0, 0, 0, 0, 0, 0, 300,
        9, 0, 0, 0, 0, 0,
********************************************
********************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
********************************************
********************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ11 2010-Feb-23, SIN-BKK 2010-Feb-23, Y, 1, 0, 0, 0, 0, 300, 0,
SQ11 2010-Feb-23, SIN-BKK 2010-Feb-23, Y, 2, 0, 0, 0, 0, 300, 0,
********************************************
********************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
        (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ11 2010-Feb-23, SIN-BKK 2010-Feb-23, Y, 1, Y, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
        0, 0, 0, 0,
SQ11 2010-Feb-23, SIN-BKK 2010-Feb-23, Y, 2, M, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
        0, 0, 0, 0,
********************************************
********************************************
FlightDate: SQ11, 2010-Feb-24
********************************************
********************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
        apsed, Distance, Capacity,
SQ11 2010-Feb-24, SIN-BKK, 2010-Feb-24, 08:20:00, 2010-Feb-24, 11:00:00, 07:40:00
        , 0, -05:00:00, 6300, 0,
********************************************
********************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
        Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ11 2010-Feb-24, SIN-BKK 2010-Feb-24, Y, 300, 300, 0, 0, 0, 0, 0, 0, 0, 0, 300,
        9, 0, 0, 0, 0, 0,
********************************************
********************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
********************************************
********************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
```

```
SQ11 2010-Feb-24, SIN-BKK 2010-Feb-24, Y, 1, 0, 0, 0, 0, 300, 0,
SQ11 2010-Feb-24, SIN-BKK 2010-Feb-24, Y, 2, 0, 0, 0, 0, 300, 0,
*******************************************
*******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
       (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ11 2010-Feb-24, SIN-BKK 2010-Feb-24, Y, 1, Y, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
       0, 0, 0, 0,
SQ11 2010-Feb-24, SIN-BKK 2010-Feb-24, Y, 2, M, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
       0, 0, 0, 0,
*******************************************
*******************************************
FlightDate: SQ11, 2010-Feb-25
*******************************************
*******************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
      apsed, Distance, Capacity,
SQ11 2010-Feb-25, SIN-BKK, 2010-Feb-25, 08:20:00, 2010-Feb-25, 11:00:00, 07:40:00
      , 0, -05:00:00, 6300, 0,
*******************************************
*******************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
      Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ11 2010-Feb-25, SIN-BKK 2010-Feb-25, Y, 300, 300, 0, 0, 0, 0, 0, 0, 0, 0, 300,
      9, 0, 0, 0, 0, 0,
*******************************************
*******************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ11 2010-Feb-25, SIN-BKK 2010-Feb-25, Y, 1, 0, 0, 0, 0, 300, 0,
SQ11 2010-Feb-25, SIN-BKK 2010-Feb-25, Y, 2, 0, 0, 0, 0, 300, 0,
*******************************************
*******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
       (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ11 2010-Feb-25, SIN-BKK 2010-Feb-25, Y, 1, Y, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
       0, 0, 0, 0,
SQ11 2010-Feb-25, SIN-BKK 2010-Feb-25, Y, 2, M, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
       0, 0, 0, 0,
*******************************************
*******************************************
FlightDate: SQ11, 2010-Feb-26
*******************************************
*******************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
      apsed, Distance, Capacity,
```

```
SQ11 2010-Feb-26, SIN-BKK, 2010-Feb-26, 08:20:00, 2010-Feb-26, 11:00:00, 07:40:00
     , 0, -05:00:00, 6300, 0,
*******************************************
*******************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
     Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ11 2010-Feb-26, SIN-BKK 2010-Feb-26, Y, 300, 300, 0, 0, 0, 0, 0, 0, 0, 0, 300,
     9, 0, 0, 0, 0, 0,
*******************************************
*******************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ11 2010-Feb-26, SIN-BKK 2010-Feb-26, Y, 1, 0, 0, 0, 0, 300, 0,
SQ11 2010-Feb-26, SIN-BKK 2010-Feb-26, Y, 2, 0, 0, 0, 0, 300, 0,
*******************************************
*******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
     (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ11 2010-Feb-26, SIN-BKK 2010-Feb-26, Y, 1, Y, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
     0, 0, 0, 0,
SQ11 2010-Feb-26, SIN-BKK 2010-Feb-26, Y, 2, M, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
     0, 0, 0, 0,
*******************************************
*******************************************
FlightDate: SQ11, 2010-Feb-27
*******************************************
*******************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
     apsed, Distance, Capacity,
SQ11 2010-Feb-27, SIN-BKK, 2010-Feb-27, 08:20:00, 2010-Feb-27, 11:00:00, 07:40:00
     , 0, -05:00:00, 6300, 0,
*******************************************
*******************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
     Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ11 2010-Feb-27, SIN-BKK 2010-Feb-27, Y, 300, 300, 0, 0, 0, 0, 0, 0, 0, 0, 300,
     9, 0, 0, 0, 0, 0,
*******************************************
*******************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ11 2010-Feb-27, SIN-BKK 2010-Feb-27, Y, 1, 0, 0, 0, 0, 300, 0,
```

```
SQ11 2010-Feb-27, SIN-BKK 2010-Feb-27, Y, 2, 0, 0, 0, 0, 300, 0,
********************************************
********************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
      (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ11 2010-Feb-27, SIN-BKK 2010-Feb-27, Y, 1, Y, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
SQ11 2010-Feb-27, SIN-BKK 2010-Feb-27, Y, 2, M, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
********************************************
********************************************
FlightDate: SQ11, 2010-Feb-28
********************************************
********************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
      apsed, Distance, Capacity,
SQ11 2010-Feb-28, SIN-BKK, 2010-Feb-28, 08:20:00, 2010-Feb-28, 11:00:00, 07:40:00
      , 0, -05:00:00, 6300, 0,
********************************************
********************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
      Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ11 2010-Feb-28, SIN-BKK 2010-Feb-28, Y, 300, 300, 0, 0, 0, 0, 0, 0, 0, 0, 300,
      9, 0, 0, 0, 0, 0,
********************************************
********************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
********************************************
********************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ11 2010-Feb-28, SIN-BKK 2010-Feb-28, Y, 1, 0, 0, 0, 0, 300, 0,
SQ11 2010-Feb-28, SIN-BKK 2010-Feb-28, Y, 2, 0, 0, 0, 0, 300, 0,
********************************************
********************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
      (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ11 2010-Feb-28, SIN-BKK 2010-Feb-28, Y, 1, Y, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
SQ11 2010-Feb-28, SIN-BKK 2010-Feb-28, Y, 2, M, 300 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
********************************************
********************************************
FlightDate: SQ12, 2010-Jan-15
********************************************
********************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
      apsed, Distance, Capacity,
SQ12 2010-Jan-15, SIN-HND, 2010-Jan-15, 09:20:00, 2010-Jan-15, 12:00:00, 07:40:00
```

```
       , 0, -05:00:00, 6300, 0,
*******************************************
*******************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
        Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ12 2010-Jan-15, SIN-HND 2010-Jan-15, Y, 200, 200, 2.082e+121, 5.53287e-48, 5.20
        268e-90, 0, 1.31346e-47, 1.05119e-153, 2.78986e+179, 0, 200, 9, 3.66962e-62, 1.08
        54e-71, 6.74783e-67, 6.9835e-77, 0,
*******************************************
*******************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
SegmentCabins:
-------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ12 2010-Jan-15, SIN-HND 2010-Jan-15, Y, 1, 0, 0, 0, 0, 200, 0,
SQ12 2010-Jan-15, SIN-HND 2010-Jan-15, Y, 2, 0, 0, 0, 0, 200, 0,
*******************************************
*******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
        (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ12 2010-Jan-15, SIN-HND 2010-Jan-15, Y, 1, Y13856, 200 (0), 0, 0, 0, 0, 0 (0),
        0, 0, 0, 0, 0, 0,
SQ12 2010-Jan-15, SIN-HND 2010-Jan-15, Y, 2, M, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
        0, 0, 0, 0,
*******************************************
*******************************************
FlightDate: SQ12, 2010-Jan-16
*******************************************
*******************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
        apsed, Distance, Capacity,
SQ12 2010-Jan-16, SIN-HND, 2010-Jan-16, 09:20:00, 2010-Jan-16, 12:00:00, 07:40:00
        , 0, -05:00:00, 6300, 0,
*******************************************
*******************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
        Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ12 2010-Jan-16, SIN-HND 2010-Jan-16, Y, 200, 200, 0, 0, 0, 0, 0, 0, 0, 0, 200,
        9, 2.63638e-319, 0, 0, 0, 0,
*******************************************
*******************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
SegmentCabins:
-------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ12 2010-Jan-16, SIN-HND 2010-Jan-16, Y, 1, 0, 0, 0, 0, 200, 0,
```

```
SQ12 2010-Jan-16, SIN-HND 2010-Jan-16, Y, 2, 0, 0, 0, 0, 200, 0,
*******************************************
*******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
      (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ12 2010-Jan-16, SIN-HND 2010-Jan-16, Y, 1, Y, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
SQ12 2010-Jan-16, SIN-HND 2010-Jan-16, Y, 2, M, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
*******************************************
*******************************************
FlightDate: SQ12, 2010-Jan-17
*******************************************
*******************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
      apsed, Distance, Capacity,
SQ12 2010-Jan-17, SIN-HND, 2010-Jan-17, 09:20:00, 2010-Jan-17, 12:00:00, 07:40:00
      , 0, -05:00:00, 6300, 0,
*******************************************
*******************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
      Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ12 2010-Jan-17, SIN-HND 2010-Jan-17, Y, 200, 200, 0, 0, 0, 0, 0, 0, 0, 0, 200,
      9, 2.39291e-319, 0, 0, 0, 0,
*******************************************
*******************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ12 2010-Jan-17, SIN-HND 2010-Jan-17, Y, 1, 0, 0, 0, 0, 200, 0,
SQ12 2010-Jan-17, SIN-HND 2010-Jan-17, Y, 2, 0, 0, 0, 0, 200, 0,
*******************************************
*******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
      (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ12 2010-Jan-17, SIN-HND 2010-Jan-17, Y, 1, Y, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
SQ12 2010-Jan-17, SIN-HND 2010-Jan-17, Y, 2, M, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
*******************************************
*******************************************
FlightDate: SQ12, 2010-Jan-18
*******************************************
*******************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
      apsed, Distance, Capacity,
SQ12 2010-Jan-18, SIN-HND, 2010-Jan-18, 09:20:00, 2010-Jan-18, 12:00:00, 07:40:00
```

```
      , 0, -05:00:00, 6300, 0,
*******************************************
*******************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
      Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ12 2010-Jan-18, SIN-HND 2010-Jan-18, Y, 200, 200, 0, 0, 0, 0, 0, 0, 0, 200,
      9, 2.14469e-319, 0, 0, 0, 0,
*******************************************
*******************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ12 2010-Jan-18, SIN-HND 2010-Jan-18, Y, 1, 0, 0, 0, 0, 200, 0,
SQ12 2010-Jan-18, SIN-HND 2010-Jan-18, Y, 2, 0, 0, 0, 0, 200, 0,
*******************************************
*******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
      (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ12 2010-Jan-18, SIN-HND 2010-Jan-18, Y, 1, Y, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
SQ12 2010-Jan-18, SIN-HND 2010-Jan-18, Y, 2, M, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
*******************************************
*******************************************
FlightDate: SQ12, 2010-Jan-19
*******************************************
*******************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
      apsed, Distance, Capacity,
SQ12 2010-Jan-19, SIN-HND, 2010-Jan-19, 09:20:00, 2010-Jan-19, 12:00:00, 07:40:00
      , 0, -05:00:00, 6300, 0,
*******************************************
*******************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
      Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ12 2010-Jan-19, SIN-HND 2010-Jan-19, Y, 200, 200, 0, 0, 0, 0, 0, 0, 0, 200,
      9, 0, 0, 0, 0, 0,
*******************************************
*******************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ12 2010-Jan-19, SIN-HND 2010-Jan-19, Y, 1, 0, 0, 0, 0, 200, 0,
SQ12 2010-Jan-19, SIN-HND 2010-Jan-19, Y, 2, 0, 0, 0, 0, 200, 0,
```

```
********************************************
********************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
        (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ12 2010-Jan-19, SIN-HND 2010-Jan-19, Y, 1, Y, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
        0, 0, 0, 0,
SQ12 2010-Jan-19, SIN-HND 2010-Jan-19, Y, 2, M, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
        0, 0, 0, 0,
********************************************
********************************************
FlightDate: SQ12, 2010-Jan-20
********************************************
********************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
      apsed, Distance, Capacity,
SQ12 2010-Jan-20, SIN-HND, 2010-Jan-20, 09:20:00, 2010-Jan-20, 12:00:00, 07:40:00
      , 0, -05:00:00, 6300, 0,
********************************************
********************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
        Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ12 2010-Jan-20, SIN-HND 2010-Jan-20, Y, 200, 200, 0, 0, 0, 0, 0, 0, 0, 0, 200,
      9, 0, 0, 0, 0, 0,
********************************************
********************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
********************************************
********************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ12 2010-Jan-20, SIN-HND 2010-Jan-20, Y, 1, 0, 0, 0, 0, 200, 0,
SQ12 2010-Jan-20, SIN-HND 2010-Jan-20, Y, 2, 0, 0, 0, 0, 200, 0,
********************************************
********************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
        (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ12 2010-Jan-20, SIN-HND 2010-Jan-20, Y, 1, Y, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
        0, 0, 0, 0,
SQ12 2010-Jan-20, SIN-HND 2010-Jan-20, Y, 2, M, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
        0, 0, 0, 0,
********************************************
********************************************
FlightDate: SQ12, 2010-Jan-21
********************************************
********************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
      apsed, Distance, Capacity,
SQ12 2010-Jan-21, SIN-HND, 2010-Jan-21, 09:20:00, 2010-Jan-21, 12:00:00, 07:40:00
      , 0, -05:00:00, 6300, 0,
```

```
*******************************************
*******************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
       Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ12 2010-Jan-21, SIN-HND 2010-Jan-21, Y, 200, 200, 0, 0, 0, 0, 0, 0, 0, 0, 200,
       9, 0, 0, 0, 0, 0,
*******************************************
*******************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ12 2010-Jan-21, SIN-HND 2010-Jan-21, Y, 1, 0, 0, 0, 0, 200, 0,
SQ12 2010-Jan-21, SIN-HND 2010-Jan-21, Y, 2, 0, 0, 0, 0, 200, 0,
*******************************************
*******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
       (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ12 2010-Jan-21, SIN-HND 2010-Jan-21, Y, 1, Y, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
       0, 0, 0, 0,
SQ12 2010-Jan-21, SIN-HND 2010-Jan-21, Y, 2, M, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
       0, 0, 0, 0,
*******************************************
*******************************************
FlightDate: SQ12, 2010-Jan-22
*******************************************
*******************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
      apsed, Distance, Capacity,
SQ12 2010-Jan-22, SIN-HND, 2010-Jan-22, 09:20:00, 2010-Jan-22, 12:00:00, 07:40:00
      , 0, -05:00:00, 6300, 0,
*******************************************
*******************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
       Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ12 2010-Jan-22, SIN-HND 2010-Jan-22, Y, 200, 200, 0, 0, 0, 0, 0, 0, 0, 0, 200,
      9, 0, 0, 0, 0, 0,
*******************************************
*******************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ12 2010-Jan-22, SIN-HND 2010-Jan-22, Y, 1, 0, 0, 0, 0, 200, 0,
SQ12 2010-Jan-22, SIN-HND 2010-Jan-22, Y, 2, 0, 0, 0, 0, 200, 0,
*******************************************
```

```
*******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
        (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ12 2010-Jan-22, SIN-HND 2010-Jan-22, Y, 1, Y, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
        0, 0, 0, 0,
SQ12 2010-Jan-22, SIN-HND 2010-Jan-22, Y, 2, M, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
        0, 0, 0, 0,
*******************************************
*******************************************
FlightDate: SQ12, 2010-Jan-23
*******************************************
*******************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
        apsed, Distance, Capacity,
SQ12 2010-Jan-23, SIN-HND, 2010-Jan-23, 09:20:00, 2010-Jan-23, 12:00:00, 07:40:00
        , 0, -05:00:00, 6300, 0,
*******************************************
*******************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
        Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ12 2010-Jan-23, SIN-HND 2010-Jan-23, Y, 200, 200, 0, 0, 0, 0, 0, 0, 0, 0, 200,
        9, 0, 0, 0, 0, 0,
*******************************************
*******************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ12 2010-Jan-23, SIN-HND 2010-Jan-23, Y, 1, 0, 0, 0, 0, 200, 0,
SQ12 2010-Jan-23, SIN-HND 2010-Jan-23, Y, 2, 0, 0, 0, 0, 200, 0,
*******************************************
*******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
        (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ12 2010-Jan-23, SIN-HND 2010-Jan-23, Y, 1, Y, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
        0, 0, 0, 0,
SQ12 2010-Jan-23, SIN-HND 2010-Jan-23, Y, 2, M, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
        0, 0, 0, 0,
*******************************************
*******************************************
FlightDate: SQ12, 2010-Jan-24
*******************************************
*******************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
        apsed, Distance, Capacity,
SQ12 2010-Jan-24, SIN-HND, 2010-Jan-24, 09:20:00, 2010-Jan-24, 12:00:00, 07:40:00
        , 0, -05:00:00, 6300, 0,
*******************************************
```

```
*******************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
      Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ12 2010-Jan-24, SIN-HND 2010-Jan-24, Y, 200, 200, 0, 0, 0, 0, 0, 0, 0, 0, 200,
      9, 0, 0, 0, 0, 0,
*******************************************
*******************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ12 2010-Jan-24, SIN-HND 2010-Jan-24, Y, 1, 0, 0, 0, 0, 200, 0,
SQ12 2010-Jan-24, SIN-HND 2010-Jan-24, Y, 2, 0, 0, 0, 0, 200, 0,
*******************************************
*******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
        (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ12 2010-Jan-24, SIN-HND 2010-Jan-24, Y, 1, Y, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
SQ12 2010-Jan-24, SIN-HND 2010-Jan-24, Y, 2, M, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
*******************************************
*******************************************
FlightDate: SQ12, 2010-Jan-25
*******************************************
*******************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
      apsed, Distance, Capacity,
SQ12 2010-Jan-25, SIN-HND, 2010-Jan-25, 09:20:00, 2010-Jan-25, 12:00:00, 07:40:00
      , 0, -05:00:00, 6300, 0,
*******************************************
*******************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
      Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ12 2010-Jan-25, SIN-HND 2010-Jan-25, Y, 200, 200, 0, 0, 0, 0, 0, 0, 0, 0, 200,
      9, 0, 0, 0, 0, 0,
*******************************************
*******************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ12 2010-Jan-25, SIN-HND 2010-Jan-25, Y, 1, 0, 0, 0, 0, 200, 0,
SQ12 2010-Jan-25, SIN-HND 2010-Jan-25, Y, 2, 0, 0, 0, 0, 200, 0,
*******************************************
*******************************************
```

```
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
        (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ12 2010-Jan-25, SIN-HND 2010-Jan-25, Y, 1, Y, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
        0, 0, 0, 0,
SQ12 2010-Jan-25, SIN-HND 2010-Jan-25, Y, 2, M, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
        0, 0, 0, 0,
*******************************************
*******************************************
FlightDate: SQ12, 2010-Jan-26
*******************************************
*******************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
        apsed, Distance, Capacity,
SQ12 2010-Jan-26, SIN-HND, 2010-Jan-26, 09:20:00, 2010-Jan-26, 12:00:00, 07:40:00
        , 0, -05:00:00, 6300, 0,
*******************************************
*******************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
        Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ12 2010-Jan-26, SIN-HND 2010-Jan-26, Y, 200, 200, 0, 0, 0, 0, 0, 0, 0, 0, 200,
        9, 0, 0, 0, 0, 0,
*******************************************
*******************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ12 2010-Jan-26, SIN-HND 2010-Jan-26, Y, 1, 0, 0, 0, 0, 200, 0,
SQ12 2010-Jan-26, SIN-HND 2010-Jan-26, Y, 2, 0, 0, 0, 0, 200, 0,
*******************************************
*******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
        (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ12 2010-Jan-26, SIN-HND 2010-Jan-26, Y, 1, Y, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
        0, 0, 0, 0,
SQ12 2010-Jan-26, SIN-HND 2010-Jan-26, Y, 2, M, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
        0, 0, 0, 0,
*******************************************
*******************************************
FlightDate: SQ12, 2010-Jan-27
*******************************************
*******************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
        apsed, Distance, Capacity,
SQ12 2010-Jan-27, SIN-HND, 2010-Jan-27, 09:20:00, 2010-Jan-27, 12:00:00, 07:40:00
        , 0, -05:00:00, 6300, 0,
*******************************************
*******************************************
```

```
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
       Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ12 2010-Jan-27, SIN-HND 2010-Jan-27, Y, 200, 200, 0, 0, 0, 0, 0, 0, 0, 0, 200,
       9, 0, 0, 0, 0, 0,
*******************************************
*******************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ12 2010-Jan-27, SIN-HND 2010-Jan-27, Y, 1, 0, 0, 0, 0, 200, 0,
SQ12 2010-Jan-27, SIN-HND 2010-Jan-27, Y, 2, 0, 0, 0, 0, 200, 0,
*******************************************
*******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
        (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ12 2010-Jan-27, SIN-HND 2010-Jan-27, Y, 1, Y, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
       0, 0, 0, 0,
SQ12 2010-Jan-27, SIN-HND 2010-Jan-27, Y, 2, M, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
       0, 0, 0, 0,
*******************************************
*******************************************
FlightDate: SQ12, 2010-Jan-28
*******************************************
*******************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
       apsed, Distance, Capacity,
SQ12 2010-Jan-28, SIN-HND, 2010-Jan-28, 09:20:00, 2010-Jan-28, 12:00:00, 07:40:00
       , 0, -05:00:00, 6300, 0,
*******************************************
*******************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
       Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ12 2010-Jan-28, SIN-HND 2010-Jan-28, Y, 200, 200, 0, 0, 0, 0, 0, 0, 0, 0, 200,
       9, 0, 0, 0, 0, 0,
*******************************************
*******************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ12 2010-Jan-28, SIN-HND 2010-Jan-28, Y, 1, 0, 0, 0, 0, 200, 0,
SQ12 2010-Jan-28, SIN-HND 2010-Jan-28, Y, 2, 0, 0, 0, 0, 200, 0,
*******************************************
*******************************************
Subclasses:
```

```
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
        (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ12 2010-Jan-28, SIN-HND 2010-Jan-28, Y, 1, Y, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
        0, 0, 0, 0,
SQ12 2010-Jan-28, SIN-HND 2010-Jan-28, Y, 2, M, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
        0, 0, 0, 0,
*******************************************
*******************************************
FlightDate: SQ12, 2010-Jan-29
*******************************************
*******************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
        apsed, Distance, Capacity,
SQ12 2010-Jan-29, SIN-HND, 2010-Jan-29, 09:20:00, 2010-Jan-29, 12:00:00, 07:40:00
        , 0, -05:00:00, 6300, 0,
*******************************************
*******************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
        Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ12 2010-Jan-29, SIN-HND 2010-Jan-29, Y, 200, 200, 0, 0, 0, 0, 0, 0, 0, 0, 200,
        9, 0, 0, 0, 0, 0,
*******************************************
*******************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ12 2010-Jan-29, SIN-HND 2010-Jan-29, Y, 1, 0, 0, 0, 0, 200, 0,
SQ12 2010-Jan-29, SIN-HND 2010-Jan-29, Y, 2, 0, 0, 0, 0, 200, 0,
*******************************************
*******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
        (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ12 2010-Jan-29, SIN-HND 2010-Jan-29, Y, 1, Y, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
        0, 0, 0, 0,
SQ12 2010-Jan-29, SIN-HND 2010-Jan-29, Y, 2, M, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
        0, 0, 0, 0,
*******************************************
*******************************************
FlightDate: SQ12, 2010-Jan-30
*******************************************
*******************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
        apsed, Distance, Capacity,
SQ12 2010-Jan-30, SIN-HND, 2010-Jan-30, 09:20:00, 2010-Jan-30, 12:00:00, 07:40:00
        , 0, -05:00:00, 6300, 0,
*******************************************
*******************************************
LegCabins:
```

```
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
      Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ12 2010-Jan-30, SIN-HND 2010-Jan-30, Y, 200, 200, 0, 0, 0, 0, 0, 0, 0, 0, 200,
      9, 0, 0, 0, 0, 0,
*******************************************
*******************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ12 2010-Jan-30, SIN-HND 2010-Jan-30, Y, 1, 0, 0, 0, 0, 200, 0,
SQ12 2010-Jan-30, SIN-HND 2010-Jan-30, Y, 2, 0, 0, 0, 0, 200, 0,
*******************************************
*******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
       (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ12 2010-Jan-30, SIN-HND 2010-Jan-30, Y, 1, Y, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
SQ12 2010-Jan-30, SIN-HND 2010-Jan-30, Y, 2, M, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
*******************************************
*******************************************
FlightDate: SQ12, 2010-Jan-31
*******************************************
*******************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
      apsed, Distance, Capacity,
SQ12 2010-Jan-31, SIN-HND, 2010-Jan-31, 09:20:00, 2010-Jan-31, 12:00:00, 07:40:00
      , 0, -05:00:00, 6300, 0,
*******************************************
*******************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
      Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ12 2010-Jan-31, SIN-HND 2010-Jan-31, Y, 200, 200, 0, 0, 0, 0, 0, 0, 0, 0, 200,
      9, 0, 0, 0, 0, 0,
*******************************************
*******************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ12 2010-Jan-31, SIN-HND 2010-Jan-31, Y, 1, 0, 0, 0, 0, 200, 0,
SQ12 2010-Jan-31, SIN-HND 2010-Jan-31, Y, 2, 0, 0, 0, 0, 200, 0,
*******************************************
*******************************************
Subclasses:
-----------
```

```
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
        (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ12 2010-Jan-31, SIN-HND 2010-Jan-31, Y, 1, Y, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
        0, 0, 0, 0,
SQ12 2010-Jan-31, SIN-HND 2010-Jan-31, Y, 2, M, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
        0, 0, 0, 0,
*******************************************
*******************************************
FlightDate: SQ12, 2010-Feb-01
*******************************************
*******************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
        apsed, Distance, Capacity,
SQ12 2010-Feb-01, SIN-HND, 2010-Feb-01, 09:20:00, 2010-Feb-01, 12:00:00, 07:40:00
        , 0, -05:00:00, 6300, 0,
*******************************************
*******************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
        Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ12 2010-Feb-01, SIN-HND 2010-Feb-01, Y, 200, 200, 0, 0, 0, 0, 0, 0, 0, 0, 200,
        9, 0, 0, 0, 0, 0,
*******************************************
*******************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ12 2010-Feb-01, SIN-HND 2010-Feb-01, Y, 1, 0, 0, 0, 0, 200, 0,
SQ12 2010-Feb-01, SIN-HND 2010-Feb-01, Y, 2, 0, 0, 0, 0, 200, 0,
*******************************************
*******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
        (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ12 2010-Feb-01, SIN-HND 2010-Feb-01, Y, 1, Y, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
        0, 0, 0, 0,
SQ12 2010-Feb-01, SIN-HND 2010-Feb-01, Y, 2, M, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
        0, 0, 0, 0,
*******************************************
*******************************************
FlightDate: SQ12, 2010-Feb-02
*******************************************
*******************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
        apsed, Distance, Capacity,
SQ12 2010-Feb-02, SIN-HND, 2010-Feb-02, 09:20:00, 2010-Feb-02, 12:00:00, 07:40:00
        , 0, -05:00:00, 6300, 0,
*******************************************
*******************************************
LegCabins:
----------
```

```
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
      Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ12 2010-Feb-02, SIN-HND 2010-Feb-02, Y, 200, 200, 0, 0, 0, 0, 0, 0, 0, 0, 200,
      9, 0, 0, 0, 0, 0,
*****************************************
*****************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*****************************************
*****************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ12 2010-Feb-02, SIN-HND 2010-Feb-02, Y, 1, 0, 0, 0, 0, 200, 0,
SQ12 2010-Feb-02, SIN-HND 2010-Feb-02, Y, 2, 0, 0, 0, 0, 200, 0,
*****************************************
*****************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
        (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ12 2010-Feb-02, SIN-HND 2010-Feb-02, Y, 1, Y, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
SQ12 2010-Feb-02, SIN-HND 2010-Feb-02, Y, 2, M, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
*****************************************
*****************************************
FlightDate: SQ12, 2010-Feb-03
*****************************************
*****************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
      apsed, Distance, Capacity,
SQ12 2010-Feb-03, SIN-HND, 2010-Feb-03, 09:20:00, 2010-Feb-03, 12:00:00, 07:40:00
      , 0, -05:00:00, 6300, 0,
*****************************************
*****************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
      Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ12 2010-Feb-03, SIN-HND 2010-Feb-03, Y, 200, 200, 0, 0, 0, 0, 0, 0, 0, 0, 200,
      9, 0, 0, 0, 0, 0,
*****************************************
*****************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*****************************************
*****************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ12 2010-Feb-03, SIN-HND 2010-Feb-03, Y, 1, 0, 0, 0, 0, 200, 0,
SQ12 2010-Feb-03, SIN-HND 2010-Feb-03, Y, 2, 0, 0, 0, 0, 200, 0,
*****************************************
*****************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
```

```
            (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ12 2010-Feb-03, SIN-HND 2010-Feb-03, Y, 1, Y, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
SQ12 2010-Feb-03, SIN-HND 2010-Feb-03, Y, 2, M, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
*******************************************
*******************************************
FlightDate: SQ12, 2010-Feb-04
*******************************************
*******************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
      apsed, Distance, Capacity,
SQ12 2010-Feb-04, SIN-HND, 2010-Feb-04, 09:20:00, 2010-Feb-04, 12:00:00, 07:40:00
      , 0, -05:00:00, 6300, 0,
*******************************************
*******************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
      Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ12 2010-Feb-04, SIN-HND 2010-Feb-04, Y, 200, 200, 0, 0, 0, 0, 0, 0, 0, 0, 200,
      9, 0, 0, 0, 0, 0,
*******************************************
*******************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ12 2010-Feb-04, SIN-HND 2010-Feb-04, Y, 1, 0, 0, 0, 0, 200, 0,
SQ12 2010-Feb-04, SIN-HND 2010-Feb-04, Y, 2, 0, 0, 0, 0, 200, 0,
*******************************************
*******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
      (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ12 2010-Feb-04, SIN-HND 2010-Feb-04, Y, 1, Y, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
SQ12 2010-Feb-04, SIN-HND 2010-Feb-04, Y, 2, M, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
*******************************************
*******************************************
FlightDate: SQ12, 2010-Feb-05
*******************************************
*******************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
      apsed, Distance, Capacity,
SQ12 2010-Feb-05, SIN-HND, 2010-Feb-05, 09:20:00, 2010-Feb-05, 12:00:00, 07:40:00
      , 0, -05:00:00, 6300, 0,
*******************************************
*******************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
```

```
         Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ12 2010-Feb-05, SIN-HND 2010-Feb-05, Y, 200, 200, 0, 0, 0, 0, 0, 0, 0, 0, 200,
         9, 0, 0, 0, 0, 0,
*******************************************
*******************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ12 2010-Feb-05, SIN-HND 2010-Feb-05, Y, 1, 0, 0, 0, 0, 200, 0,
SQ12 2010-Feb-05, SIN-HND 2010-Feb-05, Y, 2, 0, 0, 0, 0, 200, 0,
*******************************************
*******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
         (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ12 2010-Feb-05, SIN-HND 2010-Feb-05, Y, 1, Y, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
         0, 0, 0, 0,
SQ12 2010-Feb-05, SIN-HND 2010-Feb-05, Y, 2, M, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
         0, 0, 0, 0,
*******************************************
*******************************************
FlightDate: SQ12, 2010-Feb-06
*******************************************
*******************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
         apsed, Distance, Capacity,
SQ12 2010-Feb-06, SIN-HND, 2010-Feb-06, 09:20:00, 2010-Feb-06, 12:00:00, 07:40:00
         , 0, -05:00:00, 6300, 0,
*******************************************
*******************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
         Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ12 2010-Feb-06, SIN-HND 2010-Feb-06, Y, 200, 200, 0, 0, 0, 0, 0, 0, 0, 0, 200,
         9, 0, 0, 0, 0, 0,
*******************************************
*******************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ12 2010-Feb-06, SIN-HND 2010-Feb-06, Y, 1, 0, 0, 0, 0, 200, 0,
SQ12 2010-Feb-06, SIN-HND 2010-Feb-06, Y, 2, 0, 0, 0, 0, 200, 0,
*******************************************
*******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
         (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
```

```
SQ12 2010-Feb-06, SIN-HND 2010-Feb-06, Y, 1, Y, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
SQ12 2010-Feb-06, SIN-HND 2010-Feb-06, Y, 2, M, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
*******************************************
*******************************************
FlightDate: SQ12, 2010-Feb-07
*******************************************
*******************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
      apsed, Distance, Capacity,
SQ12 2010-Feb-07, SIN-HND, 2010-Feb-07, 09:20:00, 2010-Feb-07, 12:00:00, 07:40:00
      , 0, -05:00:00, 6300, 0,
*******************************************
*******************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
      Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ12 2010-Feb-07, SIN-HND 2010-Feb-07, Y, 200, 200, 0, 0, 0, 0, 0, 0, 0, 0, 200,
      9, 0, 0, 0, 0, 0,
*******************************************
*******************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ12 2010-Feb-07, SIN-HND 2010-Feb-07, Y, 1, 0, 0, 0, 0, 200, 0,
SQ12 2010-Feb-07, SIN-HND 2010-Feb-07, Y, 2, 0, 0, 0, 0, 200, 0,
*******************************************
*******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
      (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ12 2010-Feb-07, SIN-HND 2010-Feb-07, Y, 1, Y, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
SQ12 2010-Feb-07, SIN-HND 2010-Feb-07, Y, 2, M, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
*******************************************
*******************************************
FlightDate: SQ12, 2010-Feb-08
*******************************************
*******************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
      apsed, Distance, Capacity,
SQ12 2010-Feb-08, SIN-HND, 2010-Feb-08, 09:20:00, 2010-Feb-08, 12:00:00, 07:40:00
      , 0, -05:00:00, 6300, 0,
*******************************************
*******************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
      Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
```

```
SQ12 2010-Feb-08, SIN-HND 2010-Feb-08, Y, 200, 200, 0, 0, 0, 0, 0, 0, 0, 0, 200,
     9, 0, 0, 0, 0, 0,
*******************************************
*******************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
SegmentCabins:
-------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ12 2010-Feb-08, SIN-HND 2010-Feb-08, Y, 1, 0, 0, 0, 0, 200, 0,
SQ12 2010-Feb-08, SIN-HND 2010-Feb-08, Y, 2, 0, 0, 0, 0, 200, 0,
*******************************************
*******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
      (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ12 2010-Feb-08, SIN-HND 2010-Feb-08, Y, 1, Y, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
SQ12 2010-Feb-08, SIN-HND 2010-Feb-08, Y, 2, M, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
*******************************************
*******************************************
FlightDate: SQ12, 2010-Feb-09
*******************************************
*******************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
      apsed, Distance, Capacity,
SQ12 2010-Feb-09, SIN-HND, 2010-Feb-09, 09:20:00, 2010-Feb-09, 12:00:00, 07:40:00
      , 0, -05:00:00, 6300, 0,
*******************************************
*******************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
      Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ12 2010-Feb-09, SIN-HND 2010-Feb-09, Y, 200, 200, 0, 0, 0, 0, 0, 0, 0, 0, 200,
      9, 0, 0, 0, 0, 0,
*******************************************
*******************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
SegmentCabins:
-------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ12 2010-Feb-09, SIN-HND 2010-Feb-09, Y, 1, 0, 0, 0, 0, 200, 0,
SQ12 2010-Feb-09, SIN-HND 2010-Feb-09, Y, 2, 0, 0, 0, 0, 200, 0,
*******************************************
*******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
      (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ12 2010-Feb-09, SIN-HND 2010-Feb-09, Y, 1, Y, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
```

```
      0, 0, 0, 0,
SQ12 2010-Feb-09, SIN-HND 2010-Feb-09, Y, 2, M, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
*******************************************
*******************************************
FlightDate: SQ12, 2010-Feb-10
*******************************************
*******************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
      apsed, Distance, Capacity,
SQ12 2010-Feb-10, SIN-HND, 2010-Feb-10, 09:20:00, 2010-Feb-10, 12:00:00, 07:40:00
      , 0, -05:00:00, 6300, 0,
*******************************************
*******************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
      Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ12 2010-Feb-10, SIN-HND 2010-Feb-10, Y, 200, 200, 0, 0, 0, 0, 0, 0, 0, 0, 200,
      9, 0, 0, 0, 0, 0,
*******************************************
*******************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ12 2010-Feb-10, SIN-HND 2010-Feb-10, Y, 1, 0, 0, 0, 0, 200, 0,
SQ12 2010-Feb-10, SIN-HND 2010-Feb-10, Y, 2, 0, 0, 0, 0, 200, 0,
*******************************************
*******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
        (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ12 2010-Feb-10, SIN-HND 2010-Feb-10, Y, 1, Y, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
SQ12 2010-Feb-10, SIN-HND 2010-Feb-10, Y, 2, M, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
*******************************************
*******************************************
FlightDate: SQ12, 2010-Feb-11
*******************************************
*******************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
      apsed, Distance, Capacity,
SQ12 2010-Feb-11, SIN-HND, 2010-Feb-11, 09:20:00, 2010-Feb-11, 12:00:00, 07:40:00
      , 0, -05:00:00, 6300, 0,
*******************************************
*******************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
      Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ12 2010-Feb-11, SIN-HND 2010-Feb-11, Y, 200, 200, 0, 0, 0, 0, 0, 0, 0, 0, 200,
```

```
      9, 0, 0, 0, 0, 0,
*******************************************
*******************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ12 2010-Feb-11, SIN-HND 2010-Feb-11, Y, 1, 0, 0, 0, 0, 200, 0,
SQ12 2010-Feb-11, SIN-HND 2010-Feb-11, Y, 2, 0, 0, 0, 0, 200, 0,
*******************************************
*******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
      (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ12 2010-Feb-11, SIN-HND 2010-Feb-11, Y, 1, Y, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
SQ12 2010-Feb-11, SIN-HND 2010-Feb-11, Y, 2, M, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
*******************************************
*******************************************
FlightDate: SQ12, 2010-Feb-12
*******************************************
*******************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
      apsed, Distance, Capacity,
SQ12 2010-Feb-12, SIN-HND, 2010-Feb-12, 09:20:00, 2010-Feb-12, 12:00:00, 07:40:00
      , 0, -05:00:00, 6300, 0,
*******************************************
*******************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
      Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ12 2010-Feb-12, SIN-HND 2010-Feb-12, Y, 200, 200, 0, 0, 0, 0, 0, 0, 0, 0, 200,
      9, 0, 0, 0, 0, 0,
*******************************************
*******************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ12 2010-Feb-12, SIN-HND 2010-Feb-12, Y, 1, 0, 0, 0, 0, 200, 0,
SQ12 2010-Feb-12, SIN-HND 2010-Feb-12, Y, 2, 0, 0, 0, 0, 200, 0,
*******************************************
*******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
      (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ12 2010-Feb-12, SIN-HND 2010-Feb-12, Y, 1, Y, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
```

```
SQ12 2010-Feb-12, SIN-HND 2010-Feb-12, Y, 2, M, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
*******************************************
*******************************************
FlightDate: SQ12, 2010-Feb-13
*******************************************
*******************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
      apsed, Distance, Capacity,
SQ12 2010-Feb-13, SIN-HND, 2010-Feb-13, 09:20:00, 2010-Feb-13, 12:00:00, 07:40:00
      , 0, -05:00:00, 6300, 0,
*******************************************
*******************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
      Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ12 2010-Feb-13, SIN-HND 2010-Feb-13, Y, 200, 200, 0, 0, 0, 0, 0, 0, 0, 0, 200,
      9, 0, 0, 0, 0, 0,
*******************************************
*******************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ12 2010-Feb-13, SIN-HND 2010-Feb-13, Y, 1, 0, 0, 0, 0, 200, 0,
SQ12 2010-Feb-13, SIN-HND 2010-Feb-13, Y, 2, 0, 0, 0, 0, 200, 0,
*******************************************
*******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
      (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ12 2010-Feb-13, SIN-HND 2010-Feb-13, Y, 1, Y, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
SQ12 2010-Feb-13, SIN-HND 2010-Feb-13, Y, 2, M, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
*******************************************
*******************************************
FlightDate: SQ12, 2010-Feb-14
*******************************************
*******************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
      apsed, Distance, Capacity,
SQ12 2010-Feb-14, SIN-HND, 2010-Feb-14, 09:20:00, 2010-Feb-14, 12:00:00, 07:40:00
      , 0, -05:00:00, 6300, 0,
*******************************************
*******************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
      Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ12 2010-Feb-14, SIN-HND 2010-Feb-14, Y, 200, 200, 0, 0, 0, 0, 0, 0, 0, 0, 200,
      9, 0, 0, 0, 0, 0,
```

```
*******************************************
*******************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ12 2010-Feb-14, SIN-HND 2010-Feb-14, Y, 1, 0, 0, 0, 0, 200, 0,
SQ12 2010-Feb-14, SIN-HND 2010-Feb-14, Y, 2, 0, 0, 0, 0, 200, 0,
*******************************************
*******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
       (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ12 2010-Feb-14, SIN-HND 2010-Feb-14, Y, 1, Y, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
       0, 0, 0, 0,
SQ12 2010-Feb-14, SIN-HND 2010-Feb-14, Y, 2, M, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
       0, 0, 0, 0,
*******************************************
*******************************************
FlightDate: SQ12, 2010-Feb-15
*******************************************
*******************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
       apsed, Distance, Capacity,
SQ12 2010-Feb-15, SIN-HND, 2010-Feb-15, 09:20:00, 2010-Feb-15, 12:00:00, 07:40:00
       , 0, -05:00:00, 6300, 0,
*******************************************
*******************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
       Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ12 2010-Feb-15, SIN-HND 2010-Feb-15, Y, 200, 200, 0, 0, 0, 0, 0, 0, 0, 0, 200,
       9, 0, 0, 0, 0, 0,
*******************************************
*******************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ12 2010-Feb-15, SIN-HND 2010-Feb-15, Y, 1, 0, 0, 0, 0, 200, 0,
SQ12 2010-Feb-15, SIN-HND 2010-Feb-15, Y, 2, 0, 0, 0, 0, 200, 0,
*******************************************
*******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
       (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ12 2010-Feb-15, SIN-HND 2010-Feb-15, Y, 1, Y, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
       0, 0, 0, 0,
SQ12 2010-Feb-15, SIN-HND 2010-Feb-15, Y, 2, M, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
```

```
        0, 0, 0, 0,
********************************************
********************************************
FlightDate: SQ12, 2010-Feb-16
********************************************
********************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
       apsed, Distance, Capacity,
SQ12 2010-Feb-16, SIN-HND, 2010-Feb-16, 09:20:00, 2010-Feb-16, 12:00:00, 07:40:00
       , 0, -05:00:00, 6300, 0,
********************************************
********************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
       Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ12 2010-Feb-16, SIN-HND 2010-Feb-16, Y, 200, 200, 0, 0, 0, 0, 0, 0, 0, 0, 200,
       9, 0, 0, 0, 0, 0,
********************************************
********************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
********************************************
********************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ12 2010-Feb-16, SIN-HND 2010-Feb-16, Y, 1, 0, 0, 0, 0, 200, 0,
SQ12 2010-Feb-16, SIN-HND 2010-Feb-16, Y, 2, 0, 0, 0, 0, 200, 0,
********************************************
********************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
       (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ12 2010-Feb-16, SIN-HND 2010-Feb-16, Y, 1, Y, 200 (0), 0, 0, 0, 0 (0), 0, 0,
       0, 0, 0, 0,
SQ12 2010-Feb-16, SIN-HND 2010-Feb-16, Y, 2, M, 200 (0), 0, 0, 0, 0 (0), 0, 0,
       0, 0, 0, 0,
********************************************
********************************************
FlightDate: SQ12, 2010-Feb-17
********************************************
********************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
       apsed, Distance, Capacity,
SQ12 2010-Feb-17, SIN-HND, 2010-Feb-17, 09:20:00, 2010-Feb-17, 12:00:00, 07:40:00
       , 0, -05:00:00, 6300, 0,
********************************************
********************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
       Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ12 2010-Feb-17, SIN-HND 2010-Feb-17, Y, 200, 200, 0, 0, 0, 0, 0, 0, 0, 0, 200,
       9, 0, 0, 0, 0, 0,
********************************************
```

```
*******************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ12 2010-Feb-17, SIN-HND 2010-Feb-17, Y, 1, 0, 0, 0, 0, 200, 0,
SQ12 2010-Feb-17, SIN-HND 2010-Feb-17, Y, 2, 0, 0, 0, 0, 200, 0,
*******************************************
*******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
      (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ12 2010-Feb-17, SIN-HND 2010-Feb-17, Y, 1, Y, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
SQ12 2010-Feb-17, SIN-HND 2010-Feb-17, Y, 2, M, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
*******************************************
*******************************************
FlightDate: SQ12, 2010-Feb-18
*******************************************
*******************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
      apsed, Distance, Capacity,
SQ12 2010-Feb-18, SIN-HND, 2010-Feb-18, 09:20:00, 2010-Feb-18, 12:00:00, 07:40:00
      , 0, -05:00:00, 6300, 0,
*******************************************
*******************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
      Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ12 2010-Feb-18, SIN-HND 2010-Feb-18, Y, 200, 200, 0, 0, 0, 0, 0, 0, 0, 0, 200,
      9, 0, 0, 0, 0, 0,
*******************************************
*******************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ12 2010-Feb-18, SIN-HND 2010-Feb-18, Y, 1, 0, 0, 0, 0, 200, 0,
SQ12 2010-Feb-18, SIN-HND 2010-Feb-18, Y, 2, 0, 0, 0, 0, 200, 0,
*******************************************
*******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
      (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ12 2010-Feb-18, SIN-HND 2010-Feb-18, Y, 1, Y, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
SQ12 2010-Feb-18, SIN-HND 2010-Feb-18, Y, 2, M, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
```

```
**********************************************
**********************************************
FlightDate: SQ12, 2010-Feb-19
**********************************************
**********************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
      apsed, Distance, Capacity,
SQ12 2010-Feb-19, SIN-HND, 2010-Feb-19, 09:20:00, 2010-Feb-19, 12:00:00, 07:40:00
      , 0, -05:00:00, 6300, 0,
**********************************************
**********************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
      Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ12 2010-Feb-19, SIN-HND 2010-Feb-19, Y, 200, 200, 0, 0, 0, 0, 0, 0, 0, 200,
      9, 0, 0, 0, 0, 0,
**********************************************
**********************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
**********************************************
**********************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ12 2010-Feb-19, SIN-HND 2010-Feb-19, Y, 1, 0, 0, 0, 0, 200, 0,
SQ12 2010-Feb-19, SIN-HND 2010-Feb-19, Y, 2, 0, 0, 0, 0, 200, 0,
**********************************************
**********************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
      (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ12 2010-Feb-19, SIN-HND 2010-Feb-19, Y, 1, Y, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
SQ12 2010-Feb-19, SIN-HND 2010-Feb-19, Y, 2, M, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
**********************************************
**********************************************
FlightDate: SQ12, 2010-Feb-20
**********************************************
**********************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
      apsed, Distance, Capacity,
SQ12 2010-Feb-20, SIN-HND, 2010-Feb-20, 09:20:00, 2010-Feb-20, 12:00:00, 07:40:00
      , 0, -05:00:00, 6300, 0,
**********************************************
**********************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
      Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ12 2010-Feb-20, SIN-HND 2010-Feb-20, Y, 200, 200, 0, 0, 0, 0, 0, 0, 0, 200,
      9, 0, 0, 0, 0, 0,
**********************************************
**********************************************
```

```
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ12 2010-Feb-20, SIN-HND 2010-Feb-20, Y, 1, 0, 0, 0, 0, 200, 0,
SQ12 2010-Feb-20, SIN-HND 2010-Feb-20, Y, 2, 0, 0, 0, 0, 200, 0,
*******************************************
*******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
      (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ12 2010-Feb-20, SIN-HND 2010-Feb-20, Y, 1, Y, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
SQ12 2010-Feb-20, SIN-HND 2010-Feb-20, Y, 2, M, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
*******************************************
*******************************************
FlightDate: SQ12, 2010-Feb-21
*******************************************
*******************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
      apsed, Distance, Capacity,
SQ12 2010-Feb-21, SIN-HND, 2010-Feb-21, 09:20:00, 2010-Feb-21, 12:00:00, 07:40:00
      , 0, -05:00:00, 6300, 0,
*******************************************
*******************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
      Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ12 2010-Feb-21, SIN-HND 2010-Feb-21, Y, 200, 200, 0, 0, 0, 0, 0, 0, 0, 0, 200,
      9, 0, 0, 0, 0, 0,
*******************************************
*******************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ12 2010-Feb-21, SIN-HND 2010-Feb-21, Y, 1, 0, 0, 0, 0, 200, 0,
SQ12 2010-Feb-21, SIN-HND 2010-Feb-21, Y, 2, 0, 0, 0, 0, 200, 0,
*******************************************
*******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
      (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ12 2010-Feb-21, SIN-HND 2010-Feb-21, Y, 1, Y, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
SQ12 2010-Feb-21, SIN-HND 2010-Feb-21, Y, 2, M, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
*******************************************
```

```
*******************************************
FlightDate: SQ12, 2010-Feb-22
*******************************************
*******************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
     apsed, Distance, Capacity,
SQ12 2010-Feb-22, SIN-HND, 2010-Feb-22, 09:20:00, 2010-Feb-22, 12:00:00, 07:40:00
     , 0, -05:00:00, 6300, 0,
*******************************************
*******************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
     Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ12 2010-Feb-22, SIN-HND 2010-Feb-22, Y, 200, 200, 0, 0, 0, 0, 0, 0, 0, 0, 200,
     9, 0, 0, 0, 0, 0,
*******************************************
*******************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ12 2010-Feb-22, SIN-HND 2010-Feb-22, Y, 1, 0, 0, 0, 0, 200, 0,
SQ12 2010-Feb-22, SIN-HND 2010-Feb-22, Y, 2, 0, 0, 0, 0, 200, 0,
*******************************************
*******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
      (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ12 2010-Feb-22, SIN-HND 2010-Feb-22, Y, 1, Y, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
     0, 0, 0, 0,
SQ12 2010-Feb-22, SIN-HND 2010-Feb-22, Y, 2, M, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
     0, 0, 0, 0,
*******************************************
*******************************************
FlightDate: SQ12, 2010-Feb-23
*******************************************
*******************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
     apsed, Distance, Capacity,
SQ12 2010-Feb-23, SIN-HND, 2010-Feb-23, 09:20:00, 2010-Feb-23, 12:00:00, 07:40:00
     , 0, -05:00:00, 6300, 0,
*******************************************
*******************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
     Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ12 2010-Feb-23, SIN-HND 2010-Feb-23, Y, 200, 200, 0, 0, 0, 0, 0, 0, 0, 0, 200,
     9, 0, 0, 0, 0, 0,
*******************************************
*******************************************
Buckets:
```

```
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ12 2010-Feb-23, SIN-HND 2010-Feb-23, Y, 1, 0, 0, 0, 0, 200, 0,
SQ12 2010-Feb-23, SIN-HND 2010-Feb-23, Y, 2, 0, 0, 0, 0, 200, 0,
*******************************************
*******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
      (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ12 2010-Feb-23, SIN-HND 2010-Feb-23, Y, 1, Y, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
SQ12 2010-Feb-23, SIN-HND 2010-Feb-23, Y, 2, M, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
*******************************************
*******************************************
FlightDate: SQ12, 2010-Feb-24
*******************************************
*******************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
      apsed, Distance, Capacity,
SQ12 2010-Feb-24, SIN-HND, 2010-Feb-24, 09:20:00, 2010-Feb-24, 12:00:00, 07:40:00
      , 0, -05:00:00, 6300, 0,
*******************************************
*******************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
      Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ12 2010-Feb-24, SIN-HND 2010-Feb-24, Y, 200, 200, 0, 0, 0, 0, 0, 0, 0, 0, 200,
      9, 0, 0, 0, 0, 0,
*******************************************
*******************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ12 2010-Feb-24, SIN-HND 2010-Feb-24, Y, 1, 0, 0, 0, 0, 200, 0,
SQ12 2010-Feb-24, SIN-HND 2010-Feb-24, Y, 2, 0, 0, 0, 0, 200, 0,
*******************************************
*******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
      (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ12 2010-Feb-24, SIN-HND 2010-Feb-24, Y, 1, Y, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
SQ12 2010-Feb-24, SIN-HND 2010-Feb-24, Y, 2, M, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
      0, 0, 0, 0,
*******************************************
*******************************************
```

```
FlightDate: SQ12, 2010-Feb-25
********************************************
********************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
    apsed, Distance, Capacity,
SQ12 2010-Feb-25, SIN-HND, 2010-Feb-25, 09:20:00, 2010-Feb-25, 12:00:00, 07:40:00
    , 0, -05:00:00, 6300, 0,
********************************************
********************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
    Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ12 2010-Feb-25, SIN-HND 2010-Feb-25, Y, 200, 200, 0, 0, 0, 0, 0, 0, 0, 0, 200,
    9, 0, 0, 0, 0, 0,
********************************************
********************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
********************************************
********************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ12 2010-Feb-25, SIN-HND 2010-Feb-25, Y, 1, 0, 0, 0, 0, 200, 0,
SQ12 2010-Feb-25, SIN-HND 2010-Feb-25, Y, 2, 0, 0, 0, 0, 200, 0,
********************************************
********************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
    (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ12 2010-Feb-25, SIN-HND 2010-Feb-25, Y, 1, Y, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
    0, 0, 0, 0,
SQ12 2010-Feb-25, SIN-HND 2010-Feb-25, Y, 2, M, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
    0, 0, 0, 0,
********************************************
********************************************
FlightDate: SQ12, 2010-Feb-26
********************************************
********************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
    apsed, Distance, Capacity,
SQ12 2010-Feb-26, SIN-HND, 2010-Feb-26, 09:20:00, 2010-Feb-26, 12:00:00, 07:40:00
    , 0, -05:00:00, 6300, 0,
********************************************
********************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
    Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ12 2010-Feb-26, SIN-HND 2010-Feb-26, Y, 200, 200, 0, 0, 0, 0, 0, 0, 0, 0, 200,
    9, 0, 0, 0, 0, 0,
********************************************
********************************************
Buckets:
--------
```

```
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ12 2010-Feb-26, SIN-HND 2010-Feb-26, Y, 1, 0, 0, 0, 0, 200, 0,
SQ12 2010-Feb-26, SIN-HND 2010-Feb-26, Y, 2, 0, 0, 0, 0, 200, 0,
*******************************************
*******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
        (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ12 2010-Feb-26, SIN-HND 2010-Feb-26, Y, 1, Y, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
        0, 0, 0, 0,
SQ12 2010-Feb-26, SIN-HND 2010-Feb-26, Y, 2, M, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
        0, 0, 0, 0,
*******************************************
*******************************************
FlightDate: SQ12, 2010-Feb-27
*******************************************
*******************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
        apsed, Distance, Capacity,
SQ12 2010-Feb-27, SIN-HND, 2010-Feb-27, 09:20:00, 2010-Feb-27, 12:00:00, 07:40:00
        , 0, -05:00:00, 6300, 0,
*******************************************
*******************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
        Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ12 2010-Feb-27, SIN-HND 2010-Feb-27, Y, 200, 200, 0, 0, 0, 0, 0, 0, 0, 0, 200,
        9, 0, 0, 0, 0, 0,
*******************************************
*******************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ12 2010-Feb-27, SIN-HND 2010-Feb-27, Y, 1, 0, 0, 0, 0, 200, 0,
SQ12 2010-Feb-27, SIN-HND 2010-Feb-27, Y, 2, 0, 0, 0, 0, 200, 0,
*******************************************
*******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
        (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ12 2010-Feb-27, SIN-HND 2010-Feb-27, Y, 1, Y, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
        0, 0, 0, 0,
SQ12 2010-Feb-27, SIN-HND 2010-Feb-27, Y, 2, M, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
        0, 0, 0, 0,
*******************************************
*******************************************
FlightDate: SQ12, 2010-Feb-28
```

```
*******************************************
*******************************************
Leg-Dates:
----------
Flight, Leg, BoardDate, BoardTime, OffDate, OffTime, Date Offset, Time Offset, El
        apsed, Distance, Capacity,
SQ12 2010-Feb-28, SIN-HND, 2010-Feb-28, 09:20:00, 2010-Feb-28, 12:00:00, 07:40:00
        , 0, -05:00:00, 6300, 0,
*******************************************
*******************************************
LegCabins:
----------
Flight, Leg, Cabin, OffedCAP, PhyCAP, RgdADJ, AU, UPR, SS, Staff, WL, Group, Comm
        Space, AvPool, Avl, NAV, GAV, ACP, ETB, BidPrice,
SQ12 2010-Feb-28, SIN-HND 2010-Feb-28, Y, 200, 200, 0, 0, 0, 0, 0, 0, 0, 0, 200,
        9, 0, 0, 0, 0, 0,
*******************************************
*******************************************
Buckets:
--------
Flight, Leg, Cabin, Yield, AU/SI, SS, AV,
*******************************************
*******************************************
SegmentCabins:
--------------
Flight, Segment, Cabin, FF, Bkgs, MIN, UPR, CommSpace, AvPool, BP,
SQ12 2010-Feb-28, SIN-HND 2010-Feb-28, Y, 1, 0, 0, 0, 0, 200, 0,
SQ12 2010-Feb-28, SIN-HND 2010-Feb-28, Y, 2, 0, 0, 0, 0, 200, 0,
*******************************************
*******************************************
Subclasses:
-----------
Flight, Segment, Cabin, FF, Subclass, MIN/AU (Prot), Nego, NS%, OB%, Bkgs, GrpBks
        (pdg), StfBkgs, WLBkgs, ETB, ClassAvl, RevAvl, SegAvl,
SQ12 2010-Feb-28, SIN-HND 2010-Feb-28, Y, 1, Y, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
        0, 0, 0, 0,
SQ12 2010-Feb-28, SIN-HND 2010-Feb-28, Y, 2, M, 200 (0), 0, 0, 0, 0, 0 (0), 0, 0,
        0, 0, 0, 0,
*******************************************
```

12.6   Exploring the Predefined BOM Tree
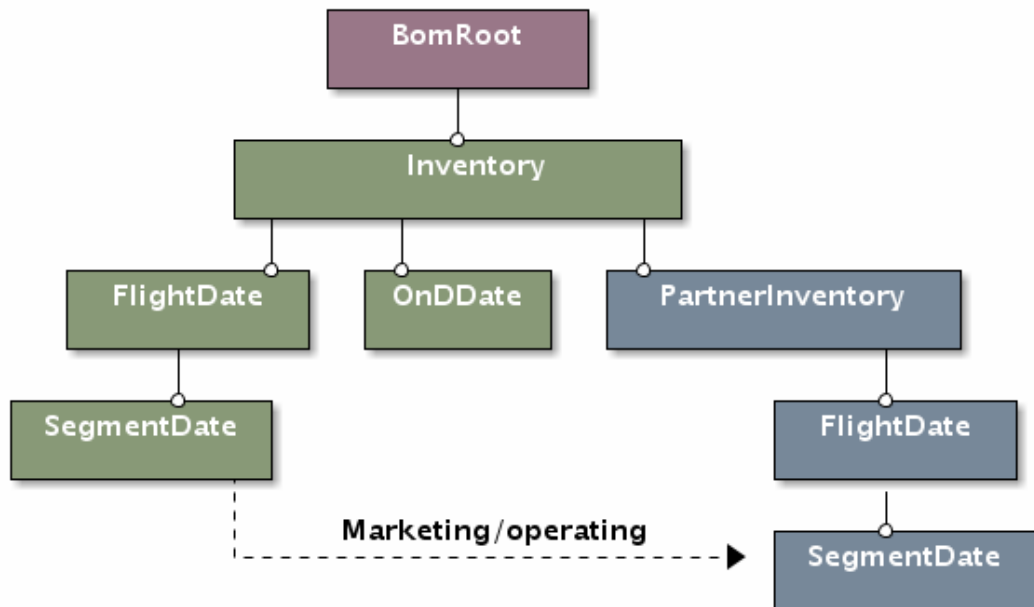
Figure 2: TravelCCM BOM tree

`TravelCCM` predefines a BOM (Business Object Model) tree specific to the airline IT arena.

12.6.1   Airline Network BOM Tree

- TRAVELCCM::ReachableUniverse

- TRAVELCCM::OriginDestinationSet

- TRAVELCCM::SegmentPathPeriod

12.6.2   Airline Schedule BOM Tree

- stdair::Inventory

- stdair::FlightPeriod

- stdair::SegmentPeriod

- stdair::OnDPeriod

12.7 Extending the BOM Tree

12.8 The travel solution calculation procedure

The project TravelCCM aims at calculating a list of `travel solutions` for every incoming `booking request`.

13 TRAVELCCM Supported Systems

13.1 Table of Contents

13.2 Introduction

This page is intended to provide a list of TRAVELCCM supported systems, i.e. the systems on which configuration, installation and testing process of the TRAVELCCM library has been sucessful. Results are grouped based on minor release number. Therefore,

only the latest tests for bug-fix releases are included. Besides, the information on this page is divided into sections dependent on the operating system.

Where necessary, some extra information is given for each tested configuration, e.g. external libraries installed, configuration commands used, etc.

If you manage to compile, install and test the TRAVELCCM library on a system not mentioned below, please let us know, so we could update this database.

13.3   TRAVELCCM 3.10.x

13.3.1   Linux Systems

13.3.1.1   Fedora Core 4 with ATLAS

- **Platform**: Intel Pentium 4

- **Operating System**: Fedora Core 4 (x86)

- **Compiler**: g++ (GCC) 4.0.2 20051125

- **TRAVELCCM release**: 3.10.0

- **External Libraries**: From FC4 distribution:

    - `fftw3.i386-3.0.1-3`
    - `fftw3-devel.i386-3.0.1-3`
    - `atlas-sse2.i386-3.6.0-8.fc4`
    - `atlas-sse2-devel.i386-3.6.0-8.fc4`
    - `blas.i386-3.0-35.fc4`
    - `lapack.i386-3.0-35.fc4`

- **Tests Status**: All tests PASSED

- **Comments**: TRAVELCCM configured with:

  ```
  % CXXFLAGS="-O3 -pipe -march=pentium4" ./configure
  ```

- **Date**: March 7, 2006

- **Tester**: Tony Ottosson

13.3.1.2   Gentoo Linux with ACML

- **Platform**: AMD Sempron 3000+

- **Operating System**: Gentoo Linux 2006.0 (x86 arch)

- **Compiler(s)**: g++ (GCC) 3.4.5

- **TRAVELCCM release**: 3.10.1

- **External Libraries**: Compiled and installed from portage tree:

    - `sci-libs/acml-3.0.0`

- **Tests Status**: All tests PASSED

- **Comments**: BLAS and LAPACK libs set by using the following system commands:

```
% eselect blas set ACML
% eselect lapack set ACML
```

   [TRAVELCCM](#) configured with:

```
% export CPPFLAGS="-I/usr/include/acml"
% ./configure --with-blas="-lblas"
```

- **Date**: March 31, 2006

- **Tester**: Adam Piatyszek (ediap)

13.3.1.3 Gentoo Linux with ATLAS

- **Platform**: Intel Pentium M Centrino

- **Operating System**: Gentoo Linux 2006.0 (x86)

- **Compiler**: g++ (GCC) 3.4.5

- **[TRAVELCCM](#) release**: 3.10.1

- **External Libraries**: Compiled and installed from portage tree:

    - `sci-libs/fftw-3.1`

    - `sci-libs/blas-atlas-3.6.0-r1`

    - `sci-libs/lapack-atlas-3.6.0`

- **Tests Status**: All tests PASSED

- **Comments**: BLAS and LAPACK libs set by using the following system commands:

```
% eselect blas set ATLAS
% eselect lapack set ATLAS
```

   [TRAVELCCM](#) configured with:

```
% ./configure --with-blas="-lblas"
```

- **Date**: March 31, 2006

- **Tester**: Adam Piatyszek (ediap)

13.3.1.4   Gentoo Linux with MKL

- **Platform**: Intel Pentium M Centrino

- **Operating System**: Gentoo Linux 2006.0 (x86 arch)

- **Compiler**: g++ (GCC) 3.4.5

- **TRAVELCCM release**: 3.10.0

- **External Libraries**: Intel Math Kernel Library (MKL) 8.0.1 installed manually in the following directory: `/opt/intel/mkl/8.0.1`

- **Tests Status**: All tests PASSED

- **Comments**: TRAVELCCM configured using the following commands:

```
% export LDFLAGS="-L/opt/intel/mkl/8.0.1/lib/32"
% export CPPFLAGS="-I/opt/intel/mkl/8.0.1/include"
% ./configure
```

- **Date**: February 28, 2006

- **Tester**: Adam Piatyszek (ediap)

13.3.1.5   Gentoo Linux with NetLib's BLAS and LAPACK

- **Platform**: Intel Pentium M Centrino

- **Operating System**: Gentoo Linux 2006.0 (x86)

- **Compiler**: g++ (GCC) 3.4.5

- **TRAVELCCM release**: 3.10.1

- **External Libraries**: Compiled and installed from portage tree:

    - `sci-libs/fftw-3.1`
    - `sci-libs/blas-reference-19940131-r2`
    - `sci-libs/cblas-reference-20030223`
    - `sci-libs/lapack-reference-3.0-r2`

- **Tests Status**: All tests PASSED

- **Comments**: BLAS and LAPACK libs set by using the following system commands:

```
% blas-config reference
% lapack-config reference
```

TRAVELCCM configured with:

```
% ./configure --with-blas="-lblas"
```

- **Date**: March 31, 2006

- **Tester**: Adam Piatyszek (ediap)

13.3.1.6   Red Hat Enterprise Linux with TRAVELCCM External

- **Platform**: Intel Pentium 4

- **Operating System**: Red Hat Enterprise Linux AS release 4 (Nahant Update 2)

- **Compiler**: g++ (GCC) 3.4.4 20050721 (Red Hat 3.4.4-2)

- **TRAVELCCM release**: 3.10.0

- **External Libraries**: BLAS, CBLAS, LAPACK and FFTW libraries from TRAVEL-CCM External 2.1.1 package

- **Tests Status**: All tests PASSED

- **Date**: March 7, 2006

- **Tester**: Erik G. Larsson

13.3.1.7   SUSE Linux 10.0 with NetLib's BLAS and LAPACK

- **Platform**: Intel Pentium 4 CPU 3.20GHz (64-bit)

- **Operating System**: SUSE Linux 10.0 (x86_64)

- **Compiler(s)**: g++ (GCC) 4.0.2

- **TRAVELCCM release**: 3.10.0

- **External Libraries**: BLAS, LAPACK and FFTW libraries installed from OpenSuse 10.0 RPM repository:

  - `blas-3.0-926`
  - `lapack-3.0-926`
  - `fftw3-3.0.1-114`
  - `fftw3-threads-3.0.1-114`
  - `fftw3-devel-3.0.1-114`

- **Tests Status**: All tests PASSED

- **Comments**: TRAVELCCM configured with:

```
% export CXXFLAGS="-m64 -march=nocona -O3 -pipe"
% ./configure --with-lapack="/usr/lib64/liblapack.so.3"
```

- **Date**: March 1, 2006

- **Tester**: Adam Piatyszek (ediap)

13.3.1.8   SUSE Linux 10.0 with MKL

- **Platform**: Intel Pentium 4 CPU 3.20GHz (64-bit)

- **Operating System**: SUSE Linux 10.0 (x86_64)

- **Compiler(s)**: g++ (GCC) 4.0.2

- **TRAVELCCM release**: 3.10.0

- **External Libraries**: Intel Math Kernel Library (MKL) 8.0.1 installed manually in the following directory: `/opt/intel/mkl/8.0.1`

- **Tests Status**: All tests PASSED

- **Comments**: TRAVELCCM configured with:

```
% export CXXFLAGS="-m64 -march=nocona -O3 -pipe"
% export LDFLAGS="-L/opt/intel/mkl/8.0.1/lib/em64t"
% export CPPFLAGS="-I/opt/intel/mkl/8.0.1/include"
% ./configure
```

- **Date**: March 1, 2006

- **Tester**: Adam Piatyszek (ediap)

13.3.2   Windows Systems

13.3.2.1   Microsoft Windows XP with Cygwin

- **Platform**: AMD Sempron 3000+

- **Operating System**: Microsoft Windows XP SP2, Cygwin 1.5.19-4

- **Compiler(s)**: g++ (GCC) 3.4.4 (cygming special)

- **TRAVELCCM release**: 3.10.1

- **External Libraries**: Installed from Cygwin's repository:

    - `fftw-3.0.1-2`
    - `fftw-dev-3.0.1-1`
    - `lapack-3.0-4`

- **Tests Status**: All tests PASSED

- **Comments**: Only static library can be built. TRAVELCCM configured with:

```
% ./configure
```

- **Date**: March 31, 2006

- **Tester**: Adam Piatyszek (ediap)

13.3.2.2   Microsoft Windows XP with Cygwin and ATLAS

- **Platform**: AMD Sempron 3000+

- **Operating System**: Microsoft Windows XP SP2, Cygwin 1.5.19-4

- **Compiler(s)**: g++ (GCC) 3.4.4 (cygming special)

- **TRAVELCCM release**: 3.10.1

- **External Libraries**: Installed from Cygwin's repository:

    - `fftw-3.0.1-2`
    - `fftw-dev-3.0.1-1`

    ATLAS BLAS and LAPACK libraries from TRAVELCCM External 2.1.1 package configured using:

    ```
    % ./configure --enable-atlas --disable-fftw
    ```

- **Tests Status**: All tests PASSED

- **Comments**: Only static library can be built. TRAVELCCM configured with:

    ```
    % export LDFLAGS="-L/usr/local/lib"
    % ./configure
    ```

- **Date**: March 31, 2006

- **Tester**: Adam Piatyszek (ediap)

13.3.2.3   Microsoft Windows XP with Cygwin and ACML

- **Platform**: AMD Sempron 3000+

- **Operating System**: Microsoft Windows XP SP2, Cygwin 1.5.19-4

- **Compiler(s)**: g++ (GCC) 3.4.4 (cygming special)

- **TRAVELCCM release**: 3.10.2

- **External Libraries**: ACML version 3.1.0 (`acml3.1.0-32-win32-g77.exe`) installed into a default directory, i.e. `"c:\Program Files\AMD\acml3.1.0"`

- **Tests Status**: All tests PASSED

- **Comments**: Only static library can be built. TRAVELCCM configured with:

    ```
    % export LDFLAGS="-L/cygdrive/c/Progra~1/AMD/acml3.1.0/gnu32/lib"
    % export CPPFLAGS="-I/cygdrive/c/Progra~1/AMD/acml3.1.0/gnu32/include"
    % ./configure --enable-debug
    ```

- **Date**: May 15, 2006

- **Tester**: Adam Piatyszek (ediap)

13.3.2.4   Microsoft Windows XP with MinGW, MSYS and ACML

- **Platform**: AMD Sempron 3000+

- **Operating System**: Microsoft Windows XP SP2, MinGW 5.0.2, MSYS 1.0.10

- **Compiler(s)**: g++ (GCC) 3.4.4 (mingw special)

- **TRAVELCCM release**: 3.10.2

- **External Libraries**: ACML version 3.1.0 (`acml3.1.0-32-win32-g77.exe`) installed into a default directory, i.e. `"c:\Program Files\AMD\acml3.1.0"`

- **Tests Status**: All tests PASSED

- **Comments**: Only static library can be built. TRAVELCCM configured with:

```
% export LDFLAGS="-L/c/Progra~1/AMD/acml3.1.0/gnu32/lib"
% export CPPFLAGS="-I/c/Progra~1/AMD/acml3.1.0/gnu32/include"
% ./configure --enable-debug
```

- **Date**: May 15, 2006

- **Tester**: Adam Piatyszek (ediap)

13.3.2.5   Microsoft Windows XP with MinGW, MSYS and TRAVELCCM External

- **Platform**: AMD Sempron 3000+

- **Operating System**: Microsoft Windows XP SP2, MinGW 5.0.2, MSYS 1.0.10

- **Compiler(s)**: g++ (GCC) 3.4.4 (mingw special)

- **TRAVELCCM release**: 3.10.5

- **External Libraries**: BLAS, CBLAS, LAPACK and FFTW libraries from TRAVEL-CCM External 2.2.0 package

- **Tests Status**: All tests PASSED

- **Comments**: Only static library can be built. TRAVELCCM configured with:

```
% export LDFLAGS="-L/usr/local/lib"
% export CPPFLAGS="-I/usr/local/include"
% export CXXFLAGS="-Wall -O3 -march=athlon-tbird -pipe"
% ./configure --disable-html-doc
```

- **Date**: August 11, 2006

- **Tester**: Adam Piatyszek (ediap)

13.3.2.6   Microsoft Windows XP with MS Visual C++ and Intel MKL

- **Platform**: AMD Sempron 3000+

- **Operating System**: Microsoft Windows XP SP2

- **Compiler(s)**: Microsoft Visual C++ 2005 .NET

- **TRAVELCCM release**: 3.10.5

- **External Libraries**: Intel Math Kernel Library (MKL) 8.1 installed manually in the following directory: `"C:\Program Files\Intel\MKL\8.1"`

- **Tests Status**: Not fully tested. Some TRAVELCCM based programs compiled and run with success.

- **Comments**: Only static library can be built. TRAVELCCM built by opening the `"win32\travelccm.vcproj"` project file in MSVC++ and executing `"Build -> Build Solution"` command from menu.

- **Date**: August 11, 2006

- **Tester**: Adam Piatyszek (ediap)

13.3.3   Unix Systems

13.3.3.1   SunOS 5.9 with TRAVELCCM External

- **Platform**: SUNW, Sun-Blade-100 (SPARC)

- **Operating System**: SunOS 5.9 Generic_112233-10

- **Compiler(s)**: g++ (GCC) 3.4.5

- **TRAVELCCM release**: 3.10.2

- **External Libraries**: BLAS, CBLAS, LAPACK and FFTW libraries from TRAVEL-CCM External 2.1.1 package. The following configuration command has been used:

```
% export CFLAGS="-mcpu=ultrasparc -O2 -pipe -funroll-all-loops"
% ./configure
```

- **Tests Status**: All tests PASSED

- **Comments**: TRAVELCCM configured with:

```
% export LDFLAGS="-L/usr/local/lib"
% export CPPFLAGS="-I/usr/local/include"
% export CXXFLAGS="-mcpu=ultrasparc -O2 -pipe"
% ./configure --enable-debug
```

- **Date**: May 15, 2006

- **Tester**: Adam Piatyszek (ediap)

14 TRAVELCCM Supported Systems (Previous Releases)

14.1 TRAVELCCM 3.9.1

14.2 TRAVELCCM 3.9.0

14.3 TRAVELCCM 3.8.1

15 Tutorials

15.1 Table of Contents

- Preparing the TravelCCM Project for Development
- Your first networkBuilde
    - Summary of the different steps
    - Result of the Batch Program
- Network building with an input file
    - How to build a network input file?
    - Building the BOM tree with an input file
    - Result of the Batch Program

15.2 Preparing the TravelCCM Project for Development

The source code for these examples can be found in the `batches` and `test/travelccm` directories. They are compiled along with the rest of the `TravelCCM` project. See the Users Guide for more details on how to build the `TravelCCM` project.

15.3 Your first networkBuilde

15.3.1 Summary of the different steps

All the steps below can be found in the same order in the batch `TravelCCM.cpp` program.

First, we instanciate the TRAVELCCM_Service object:

Then, we construct a default sample list of travel solutions and a default booking request (as mentionned in ug_procedure_bookingrequest and ug_procedure_travelsolution parts):

```
stdair::TravelSolutionList_T lTSList;
```

For basic use, the default BOM tree can be built using:

The main step is the network building (see The travel solution calculation procedure):

15.3.2 Result of the Batch Program

When the `TravelCCM.cpp` program is run (with the `-b` option), the log output file should look like:

What is interesting is to compare the travel solution list (here reduced to a single travel solution) displayed before:

and after the network building:

Between the two groups of dashes, we can see that a network option structure has been added by the network builder: the price is 450 EUR for the Y class, the ticket is refundable but there are exchange fees and the customer must stay over on saturday night.

Let's return to our default BOM tree display: the only network rule stored was a match for the travel solution into consideration (same origin airport, same destination airport, flight date included in the network rule date range, same airline "BA", ...).

By looking at the network rule trip type "RT", we can guess we face a round trip network: that means the price given in the default bom tree construction in `stdair::CmdBomManager.hpp` has been divided by 2 because we are considering either an inbound trip or an outbound one.

15.4 Network building with an input file

15.4.1 How to build a network input file?

The objective here is to build a network input file to network builde the default travel solution list built using:

This travel solution list, reduced to a singleton, can be displayed as done before:

We deduce:

- we need a network rule whose origin-destination couple is "LHR, SYD".

- the date range must include the date "2011-06-10".

- the time range must include the time "21:45".

- the airline operating is "BA", so it must be the airline pricing.

We can deduce a part of our network rule file :

```
// Fares: fare ID; OriginCity; DestinationCity; TripType; DateRangeStart; DateRan
    geEnd; DepartureTimeRangeStart; DepartureTimeRangeEnd; POS; CabinCode; Channel; A
    dvancePurchase; SaturdayNight; ChangeFees; NonRefundable; MinimumStay; Price; nb
    Segments
// Segment: AirlineCode; Class;
1; LHR; SYD; ??; 2011-01-01; 2011-12-31; 00:00; 23:59; ???; ?; ??; ?; ?; ?; ?; ?;
    ????; BA; ?;
```

We have no information about stay duration and advance purchase (such information are contained into the booking request): so let us put "0" to embrace all the requests possible.

No information for the point-of-sale and the channel too: let us consider all the channels ("IN", "DN", "IF" and DF") and all the points of sale (the origin "LHR", the destination "SYD" and the rest-of-the-world "ROW") existing. To access this information, we could look into the default booking request.

The input file is now:

```
// Fares: fare ID; OriginCity; DestinationCity; TripType; DateRangeStart; DateRan
    geEnd; DepartureTimeRangeStart; DepartureTimeRangeEnd; POS; CabinCode; Channel; A
    dvancePurchase; SaturdayNight; ChangeFees; NonRefundable; MinimumStay; Price; nb
    Segments
// Segment: AirlineCode; Class;
1; LHR; SYD; ??; 2011-01-01; 2011-12-31; 00:00; 23:59; LHR; ?; IN; 0; ?; ?; ?; 0;
    ????; BA; ?;
2; LHR; SYD; ??; 2011-01-01; 2011-12-31; 00:00; 23:59; LHR; ?; IF; 0; ?; ?; ?; 0;
    ????; BA; ?;
3; LHR; SYD; ??; 2011-01-01; 2011-12-31; 00:00; 23:59; LHR; ?; DN; 0; ?; ?; ?; 0;
    ????; BA; ?;
4; LHR; SYD; ??; 2011-01-01; 2011-12-31; 00:00; 23:59; LHR; ?; DF; 0; ?; ?; ?; 0;
    ????; BA; ?;
5; LHR; SYD; ??; 2011-01-01; 2011-12-31; 00:00; 23:59; SYD; ?; IN; 0; ?; ?; ?; 0;
    ????; BA; ?;
6; LHR; SYD; ??; 2011-01-01; 2011-12-31; 00:00; 23:59; SYD; ?; IF; 0; ?; ?; ?; 0;
    ????; BA; ?;
7; LHR; SYD; ??; 2011-01-01; 2011-12-31; 00:00; 23:59; SYD; ?; DN; 0; ?; ?; ?; 0;
    ????; BA; ?;
8; LHR; SYD; ??; 2011-01-01; 2011-12-31; 00:00; 23:59; SYD; ?; DF; 0; ?; ?; ?; 0;
    ????; BA; ?;
9; LHR; SYD; ??; 2011-01-01; 2011-12-31; 00:00; 23:59; ROW; ?; IN; 0; ?; ?; ?; 0;
    ????; BA; ?;
10; LHR; SYD; ??; 2011-01-01; 2011-12-31; 00:00; 23:59; ROW; ?; IF; 0; ?; ?; ?; 0
```

```
    ; ????; BA; ?;
11; LHR; SYD; ??; 2011-01-01; 2011-12-31; 00:00; 23:59; ROW; ?; DN; 0; ?; ?; ?; 0
    ; ????; BA; ?;
12; LHR; SYD; ??; 2011-01-01; 2011-12-31; 00:00; 23:59; ROW; ?; DF; 0; ?; ?; ?; 0
    ; ????; BA; ?;
```

Let us say we have just the Economy cabin "Y" and Bristish Airways prices ticket for class "Y".

No information about the trip type, so we duplicate all the network rules for both type: one-way "OW" and round-trip "RT" (to access this information, we could look to the default booking request).

The network options are all set to a default value "T" (meaning true) and the network values are chosen to be all distinct.

We obtain:

```
// Fares: fare ID; OriginCity; DestinationCity; TripType; DateRangeStart; DateRan
    geEnd; DepartureTimeRangeStart; DepartureTimeRangeEnd; POS; CabinCode; Channel; A
    dvancePurchase; SaturdayNight; ChangeFees; NonRefundable; MinimumStay; Price; nb
    Segments
// Segment: AirlineCode; Class;
1; LHR; SYD; OW; 2011-01-01; 2011-12-31; 00:00; 23:59; LHR; Y; IN; 0; T; T; T; 0;
    50; BA; Y;
2; LHR; SYD; OW; 2011-01-01; 2011-12-31; 00:00; 23:59; LHR; Y; IF; 0; T; T; T; 0;
    150; BA; Y;
3; LHR; SYD; OW; 2011-01-01; 2011-12-31; 00:00; 23:59; LHR; Y; DN; 0; T; T; T; 0;
    250; BA; Y;
4; LHR; SYD; OW; 2011-01-01; 2011-12-31; 00:00; 23:59; LHR; Y; DF; 0; T; T; T; 0;
    350; BA; Y;
5; LHR; SYD; OW; 2011-01-01; 2011-12-31; 00:00; 23:59; SYD; Y; IN; 0; T; T; T; 0;
    450; BA; Y;
6; LHR; SYD; OW; 2011-01-01; 2011-12-31; 00:00; 23:59; SYD; Y; IF; 0; T; T; T; 0;
    550; BA; Y;
7; LHR; SYD; OW; 2011-01-01; 2011-12-31; 00:00; 23:59; SYD; Y; DN; 0; T; T; T; 0;
    650; BA; Y;
8; LHR; SYD; OW; 2011-01-01; 2011-12-31; 00:00; 23:59; SYD; Y; DF; 0; T; T; T; 0;
    750; BA; Y;
9; LHR; SYD; OW; 2011-01-01; 2011-12-31; 00:00; 23:59; ROW; Y; IN; 0; T; T; T; 0;
    850; BA; Y;
10; LHR; SYD; OW; 2011-01-01; 2011-12-31; 00:00; 23:59; ROW; Y; IF; 0; T; T; T; 0
    ; 950; BA; Y;
11; LHR; SYD; OW; 2011-01-01; 2011-12-31; 00:00; 23:59; ROW; Y; DN; 0; T; T; T; 0
    ; 1050; BA; Y;
12; LHR; SYD; OW; 2011-01-01; 2011-12-31; 00:00; 23:59; ROW; Y; DF; 0; T; T; T; 0
    ; 1150; BA; Y;
13; LHR; SYD; RT; 2011-01-01; 2011-12-31; 00:00; 23:59; LHR; Y; IN; 0; T; T; T; 0
    ; 90; BA; Y;
14; LHR; SYD; RT; 2011-01-01; 2011-12-31; 00:00; 23:59; LHR; Y; IF; 0; T; T; T; 0
    ; 190; BA; Y;
15; LHR; SYD; RT; 2011-01-01; 2011-12-31; 00:00; 23:59; LHR; Y; DN; 0; T; T; T; 0
    ; 290; BA; Y;
16; LHR; SYD; RT; 2011-01-01; 2011-12-31; 00:00; 23:59; LHR; Y; DF; 0; T; T; T; 0
    ; 390; BA; Y;
17; LHR; SYD; RT; 2011-01-01; 2011-12-31; 00:00; 23:59; SYD; Y; IN; 0; T; T; T; 0
    ; 490; BA; Y;
18; LHR; SYD; RT; 2011-01-01; 2011-12-31; 00:00; 23:59; SYD; Y; IF; 0; T; T; T; 0
    ; 590; BA; Y;
19; LHR; SYD; RT; 2011-01-01; 2011-12-31; 00:00; 23:59; SYD; Y; DN; 0; T; T; T; 0
    ; 690; BA; Y;
```

```
20; LHR; SYD; RT; 2011-01-01; 2011-12-31; 00:00; 23:59; SYD; Y; DF; 0; T; T; T; 0
    ; 790; BA; Y;
21; LHR; SYD; RT; 2011-01-01; 2011-12-31; 00:00; 23:59; ROW; Y; IN; 0; T; T; T; 0
    ; 890; BA; Y;
22; LHR; SYD; RT; 2011-01-01; 2011-12-31; 00:00; 23:59; ROW; Y; IF; 0; T; T; T; 0
    ; 990; BA; Y;
23; LHR; SYD; RT; 2011-01-01; 2011-12-31; 00:00; 23:59; ROW; Y; DN; 0; T; T; T; 0
    ; 1090; BA; Y;
24; LHR; SYD; RT; 2011-01-01; 2011-12-31; 00:00; 23:59; ROW; Y; DF; 0; T; T; T; 0
    ; 1190; BA; Y;
```

15.4.2   Building the BOM tree with an input file

The steps are the same as before Summary of the different steps except the bom tree
must be built using the network input file :

15.4.3   Result of the Batch Program

When the `TravelCCM.cpp` program is run with the `-f` option linking with the file built
just above:

```
~/TravelCCM -f ~/<YourFileName>.csv
```

the last lines of the log output should look like:

```
[D]~/TravelCCMgit/TravelCCM/batches/TravelCCM.cpp:223: Travel solutions:
    [0] [0] BA, 9, 2011-06-10, LHR, SYD, 21:45 --- Y, 145, 1 1 1 ---
```

We have just one network option added to the travel solution. We can deduce from the
price value 145 that the network builder used the network rule number 15 to price the
travel solution. We have an inbound or outbound trip of a round trip: the total price 290
has been divided by 2.

16   Command-Line Test to Demonstrate How To Test the Travel CCM Project

```
 */
// //////////////////////////////////////////////////////////////////
// Import section
// //////////////////////////////////////////////////////////////////
// STL
#include <sstream>
#include <fstream>
#include <string>
// Boost Unit Test Framework (UTF)
#define BOOST_TEST_DYN_LINK
#define BOOST_TEST_MAIN
#define BOOST_TEST_MODULE TravelCCMTest
#include <boost/test/unit_test.hpp>
```

```cpp
// StdAir
#include <stdair/basic/BasLogParams.hpp>
#include <stdair/basic/BasDBParams.hpp>
#include <stdair/basic/BasFileMgr.hpp>
#include <stdair/bom/TravelSolutionStruct.hpp>
#include <stdair/bom/BookingRequestStruct.hpp>
#include <stdair/service/Logger.hpp>
// TravelCCM
#include <travelccm/TRAVELCCM_Service.hpp>
#include <travelccm/config/travelccm-paths.hpp>

namespace boost_utf = boost::unit_test;

// (Boost) Unit Test XML Report
std::ofstream utfReportStream ("TravelChoiceTestSuite_utfresults.xml");

struct UnitTestConfig {
  UnitTestConfig() {
    boost_utf::unit_test_log.set_stream (utfReportStream);
    boost_utf::unit_test_log.set_format (boost_utf::XML);
    boost_utf::unit_test_log.set_threshold_level (boost_utf::log_test_units);
    //boost_utf::unit_test_log.set_threshold_level (boost_utf::log_successful_tes
      ts);
  }
  ~UnitTestConfig() {
  }
};


// /////////////// Main: Unit Test Suite ///////////////

// Set the UTF configuration (re-direct the output to a specific file)
BOOST_GLOBAL_FIXTURE (UnitTestConfig);

// Start the test suite
BOOST_AUTO_TEST_SUITE (master_test_suite)


BOOST_AUTO_TEST_CASE (simple_simulation_test) {

  // Input file name
  /*
    const stdair::Filename_T inputFileName (STDAIR_SAMPLE_DIR "/ccm_02.csv");

  // Check that the file path given as input corresponds to an actual file
  const bool doesExistAndIsReadable =
    stdair::BasFileMgr::doesExistAndIsReadable (lInputFilename);
  BOOST_CHECK_MESSAGE (doesExistAndIsReadable == true,
                       "The '" << lInputFilename
                       << "' input file can not be open and read");
  */

  // Output log File
  const stdair::Filename_T lLogFilename ("TravelChoiceTestSuite.log");

  // Set the log parameters
  std::ofstream logOutputFile;
  // Open and clean the log outputfile
  logOutputFile.open (lLogFilename.c_str());
  logOutputFile.clear();

  // Initialise the service context
```

```
  const stdair::BasLogParams lLogParams (stdair::LOG::DEBUG, logOutputFile);

  // Build the BOM tree
  TRAVELCCM::TRAVELCCM_Service travelccmService (lLogParams);

  // DEBUG
  STDAIR_LOG_DEBUG ("Welcome to TravelCCM");

  // Build a list of travel solutions
  const stdair::BookingRequestStruct& lBookingRequest =
    travelccmService.buildSampleBookingRequest();

  // DEBUG
  STDAIR_LOG_DEBUG ("Booking request: " << lBookingRequest.display());

  // Build the sample BOM tree
  stdair::TravelSolutionList_T lTSList;
  travelccmService.buildSampleTravelSolutions (lTSList);

  // DEBUG: Display the list of travel solutions
  const std::string& lCSVDump = travelccmService.csvDisplay (lTSList);
  STDAIR_LOG_DEBUG (lCSVDump);

  // Choose a travel solution
  const stdair::TravelSolutionStruct* lTS_ptr =
    travelccmService.chooseTravelSolution (lTSList, lBookingRequest);

  // Check that a solution has been found
  BOOST_REQUIRE_MESSAGE (lTS_ptr != NULL,
                         "No travel solution can be found for "
                         << lBookingRequest.display()
                         << " within the following list of travel solutions. "
                         << lCSVDump);

  // Close the log file
  logOutputFile.close();
}

// End the test suite
BOOST_AUTO_TEST_SUITE_END()

/*!
```

17 Directory Hierarchy

17.1 Directories

This directory hierarchy is sorted roughly, but not completely, alphabetically:

18 Namespace Index

18.1 Namespace List

Here is a list of all namespaces with brief descriptions:

19 Class Index

19.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

std::basic_fstream< char >
std::basic_fstream< wchar_t >
std::basic_ifstream< char >
std::basic_ifstream< wchar_t >
std::basic_ios< char >
std::basic_ios< wchar_t >
std::basic_iostream< char >
std::basic_iostream< wchar_t >
std::basic_istream< char >
std::basic_istream< wchar_t >
std::basic_istringstream< char >
std::basic_istringstream< wchar_t >
std::basic_ofstream< char >
std::basic_ofstream< wchar_t >
std::basic_ostream< char >
std::basic_ostream< wchar_t >
std::basic_ostringstream< char >
std::basic_ostringstream< wchar_t >
std::basic_string< char >

20 Class Index

20.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

21   File Index

21.1   File List

Here is a list of all files with brief descriptions:

22 Directory Documentation

22.1 travelccm/basic/ Directory Reference

Files

- file BasConst.cpp
- file BasConst_General.hpp

22.2 travelccm/batches/ Directory Reference

Files

- file travelccm.cpp

22.3 travelccm/bom/ Directory Reference

Files

- file PriceOrientedModel.cpp
- file PriceOrientedModel.hpp

22.4 travelccm/command/ Directory Reference

Files

- file ChoiceManager.cpp
- file ChoiceManager.hpp
- file CmdAbstract.cpp
- file CmdAbstract.hpp
- file FileMgr.cpp
- file FileMgr.hpp

22.5   travelccm/config/ Directory Reference

Files

- file travelccm-paths.hpp
- file travelccm-paths.hpp.in

22.6   travelccm/factory/ Directory Reference

Files

- file FacTRAVELCCMServiceContext.cpp
- file FacTRAVELCCMServiceContext.hpp

22.7   travelccm/service/ Directory Reference

Files

- file TRAVELCCM_Service.cpp
- file TRAVELCCM_ServiceContext.cpp
- file TRAVELCCM_ServiceContext.hpp

22.8   test/ Directory Reference

Directories

- directory travelccm

22.9   travelccm/ Directory Reference

Directories

- directory basic
- directory batches
- directory bom
- directory command
- directory config
- directory factory
- directory service

Files

- file TRAVELCCM_Service.hpp
- file TRAVELCCM_Types.hpp

22.10   test/travelccm/ Directory Reference

Files

- file TravelChoiceTestSuite.cpp
- file TravelChoiceTestSuite.hpp

23   Namespace Documentation

23.1   stdair Namespace Reference

Forward declarations.

23.1.1   Detailed Description

Forward declarations.

23.2   TRAVELCCM Namespace Reference

Classes

- class PriceOrientedModel
- class ChoiceManager
- class CmdAbstract
- class FileMgr
- class FacTRAVELCCMServiceContext
- class TRAVELCCM_ServiceContext

    Inner class holding the context for the TravelCCM Service object.

- class TRAVELCCM_Service
- class CustomerChoiceException

Typedefs

- typedef boost::shared_ptr< TRAVELCCM_Service > TRAVELCCM_ServicePtr_-
  T

23.2.1   Typedef Documentation

23.2.1.1   typedef boost::shared_ptr<**TRAVELCCM_Service**> **TRAVELCCM::TRAVELCCM_ServicePtr_T**

Pointer on the TRAVELCCM Service handler.

Definition at line 38 of file TRAVELCCM_Types.hpp.

24 Class Documentation

24.1 TRAVELCCM::ChoiceManager Class Reference

```
#include <travelccm/command/ChoiceManager.hpp>
```

Static Public Member Functions

- static const stdair::TravelSolutionStruct ∗ chooseTravelSolution (stdair::TravelSolutionList_-
  T &, const stdair::BookingRequestStruct &)

24.1.1 Detailed Description

Class managing the customer choice model/algorithm.

Definition at line 21 of file ChoiceManager.hpp.

24.1.2 Member Function Documentation

24.1.2.1 const stdair::TravelSolutionStruct ∗ TRAVEL-CCM::ChoiceManager::chooseTravelSolution (stdair::TravelSolutionList_T & *ioTravelSolutionList,* const stdair::BookingRequestStruct & *iBookingRequest*) `[static]`

Choose the travel solution and the fare option within the given list of travel solutions.

The returned pointer will be NULL if no travel solution is chosen (e.g. Willingness-To-Pay too low).

Definition at line 17 of file ChoiceManager.cpp.

The documentation for this class was generated from the following files:

- travelccm/command/ChoiceManager.hpp
- travelccm/command/ChoiceManager.cpp

24.2 TRAVELCCM::CmdAbstract Class Reference

```
#include <travelccm/command/CmdAbstract.hpp>
```

24.2.1 Detailed Description

Base class for the Command layer.

Definition at line 11 of file CmdAbstract.hpp.

The documentation for this class was generated from the following file:

- travelccm/command/CmdAbstract.hpp

24.3 TRAVELCCM::CustomerChoiceException Class Reference

`#include <travelccm/TRAVELCCM_Types.hpp>`

Inheritance diagram for TRAVELCCM::CustomerChoiceException:

```
                    RootException
                         ▲
                         │
         TRAVELCCM::CustomerChoiceException
```
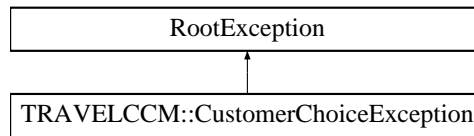
Public Member Functions

- CustomerChoiceException (const std::string &iWhat)

24.3.1 Detailed Description

TravelCCM specific root exception.

Definition at line 24 of file TRAVELCCM_Types.hpp.

24.3.2 Constructor & Destructor Documentation

24.3.2.1 TRAVELCCM::CustomerChoiceException::CustomerChoiceException (const std::string & *iWhat*) `[inline]`

Constructor.

Definition at line 29 of file TRAVELCCM_Types.hpp.

The documentation for this class was generated from the following file:

- travelccm/TRAVELCCM_Types.hpp

24.4 FacServiceAbstract Class Reference

Inheritance diagram for FacServiceAbstract:

```
                   FacServiceAbstract
                         ▲
                         │
        TRAVELCCM::FacTRAVELCCMServiceContext
```
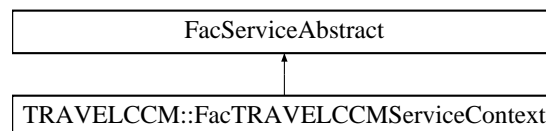
The documentation for this class was generated from the following file:

> • travelccm/factory/FacTRAVELCCMServiceContext.hpp

24.5 TRAVELCCM::FacTRAVELCCMServiceContext Class Reference

`#include <travelccm/factory/FacTRAVELCCMServiceContext.hpp>`

Inheritance diagram for TRAVELCCM::FacTRAVELCCMServiceContext:
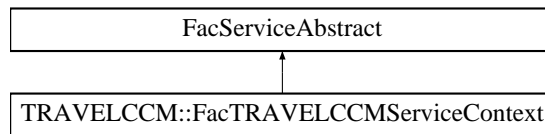
```
┌─────────────────────────────────────────────────┐
│              FacServiceAbstract                   │
└─────────────────────────────────────────────────┘
                        ▲
                        │
┌─────────────────────────────────────────────────┐
│  TRAVELCCM::FacTRAVELCCMServiceContext            │
└─────────────────────────────────────────────────┘
```

Public Member Functions

- ∼FacTRAVELCCMServiceContext ()
- TRAVELCCM_ServiceContext & create ()

Static Public Member Functions

- static FacTRAVELCCMServiceContext & instance ()

Protected Member Functions

- FacTRAVELCCMServiceContext ()

24.5.1 Detailed Description

Factory for Bucket.

Definition at line 18 of file FacTRAVELCCMServiceContext.hpp.

24.5.2 Constructor & Destructor Documentation

24.5.2.1 TRAVELCCM::FacTRAVELCCMServiceContext::∼FacTRAVELCCMServiceContext ( )

Destructor.

The Destruction put the _instance to NULL in order to be clean for the next FacTRAV-ELCCMServiceContext::instance().

Definition at line 17 of file FacTRAVELCCMServiceContext.cpp.

24.5.2.2 TRAVELCCM::FacTRAVELCCMServiceContext::FacTRAVELCCMServiceContext ( )
`[inline, protected]`

Default Constructor.

This constructor is protected in order to ensure the singleton pattern.

Definition at line 42 of file FacTRAVELCCMServiceContext.hpp.

Referenced by instance().

24.5.3 Member Function Documentation

24.5.3.1 FacTRAVELCCMServiceContext & TRAVEL-CCM::FacTRAVELCCMServiceContext::instance ( )
`[static]`

Provide the unique instance.

The singleton is instantiated when first used

Returns

> FacTRAVELCCMServiceContext&

Definition at line 22 of file FacTRAVELCCMServiceContext.cpp.

References FacTRAVELCCMServiceContext().

24.5.3.2 TRAVELCCM_ServiceContext & TRAVEL-CCM::FacTRAVELCCMServiceContext::create ( )

Create a new TRAVELCCM_ServiceContext object.

This new object is added to the list of instantiated objects.

Returns

> TRAVELCCM_ServiceContext& The newly created object.
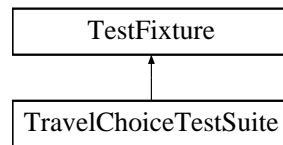
Definition at line 34 of file FacTRAVELCCMServiceContext.cpp.

The documentation for this class was generated from the following files:

- travelccm/factory/FacTRAVELCCMServiceContext.hpp
- travelccm/factory/FacTRAVELCCMServiceContext.cpp

24.6 TRAVELCCM::FileMgr Class Reference

```
#include <travelccm/command/FileMgr.hpp>
```

24.6.1   Detailed Description

Class filling the TravelSolutionHolder structure (representing a list of classes/travelSolutions) from a given input file.

Definition at line 15 of file FileMgr.hpp.

The documentation for this class was generated from the following file:

- travelccm/command/FileMgr.hpp

24.7   TRAVELCCM::PriceOrientedModel Class Reference

```
#include <travelccm/bom/PriceOrientedModel.hpp>
```

Static Public Member Functions

- static const stdair::TravelSolutionStruct ∗ chooseTravelSolution (stdair::TravelSolutionList_-T &, const stdair::BookingRequestStruct &)

24.7.1   Detailed Description

Class implementing the customer choice of cheapest solution.

Definition at line 21 of file PriceOrientedModel.hpp.

24.7.2   Member Function Documentation

**24.7.2.1   const stdair::TravelSolutionStruct ∗ TRAVEL-
CCM::PriceOrientedModel::chooseTravelSolution (
stdair::TravelSolutionList_T & *ioTSList,* const stdair::BookingRequestStruct &
iBookingRequest)** `[static]`

Choose the cheapest travel solution and the fare option within the given list of travel solutions.

The returned pointer will be NULL if no travel solution is chosen (e.g. Willingness-To-Pay too low).

Definition at line 21 of file PriceOrientedModel.cpp.

The documentation for this class was generated from the following files:

- travelccm/bom/PriceOrientedModel.hpp
- travelccm/bom/PriceOrientedModel.cpp

24.8   RootException Class Reference

Inheritance diagram for RootException:
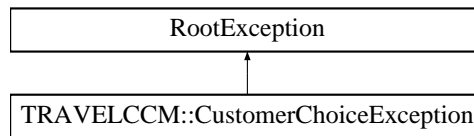
```
                        ┌──────────────────────────────┐
                        │        RootException          │
                        └──────────────────────────────┘
                                        ▲
                        ┌──────────────────────────────┐
                        │ TRAVELCCM::CustomerChoiceException │
                        └──────────────────────────────┘
```

The documentation for this class was generated from the following file:

- travelccm/TRAVELCCM_Types.hpp

24.9   ServiceAbstract Class Reference

Inheritance diagram for ServiceAbstract:

```
                        ┌──────────────────────────────┐
                        │        ServiceAbstract        │
                        └──────────────────────────────┘
                                        ▲
                        ┌──────────────────────────────┐
                        │ TRAVELCCM::TRAVELCCM_ServiceContext │
                        └──────────────────────────────┘
```
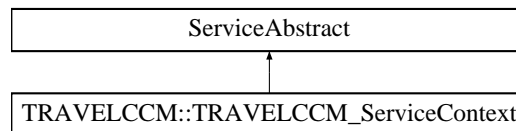
The documentation for this class was generated from the following file:

- travelccm/service/TRAVELCCM_ServiceContext.hpp

24.10   TestFixture Class Reference

Inheritance diagram for TestFixture:
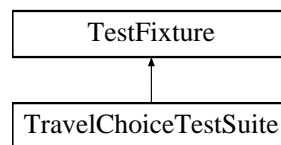
```
                        ┌──────────────────────────────┐
                        │          TestFixture          │
                        └──────────────────────────────┘
                                        ▲
                        ┌──────────────────────────────┐
                        │      TravelChoiceTestSuite     │
                        └──────────────────────────────┘
```

The documentation for this class was generated from the following file:

- test/travelccm/TravelChoiceTestSuite.hpp

24.11   TRAVELCCM::TRAVELCCM_Service Class Reference

```
#include <travelccm/TRAVELCCM_Service.hpp>
```

Public Member Functions

- TRAVELCCM_Service (const stdair::BasLogParams &, const stdair::BasDBParams &)
- TRAVELCCM_Service (const stdair::BasLogParams &)
- TRAVELCCM_Service (stdair::STDAIR_ServicePtr_T)
- ∼TRAVELCCM_Service ()
- void buildSampleBom ()
- void buildSampleTravelSolutions (stdair::TravelSolutionList_T &)
- stdair::BookingRequestStruct buildSampleBookingRequest (const bool isForCRS=false)
- const stdair::TravelSolutionStruct ∗ chooseTravelSolution (stdair::TravelSolutionList_-T &, const stdair::BookingRequestStruct &)
- std::string csvDisplay () const
- std::string csvDisplay (const stdair::TravelSolutionList_T &) const

24.11.1   Detailed Description

Interface for the TRAVELCCM Services.

Definition at line 31 of file TRAVELCCM_Service.hpp.

24.11.2   Constructor & Destructor Documentation

24.11.2.1   TRAVELCCM::TRAVELCCM_Service::TRAVELCCM_Service (const stdair::BasLogParams & *iLogParams,* const stdair::BasDBParams & *iDBParams*)

Constructor.

The init() method is called; see the corresponding documentation for more details.

A reference on an output stream is given, so that log outputs can be directed onto that stream.

Moreover, database connection parameters are given, so that a session can be created on the corresponding database.

Parameters

const	stdair::BasLogParams& Parameters for the output log stream.
const	stdair::BasDBParams& Parameters for the database access.

Definition at line 36 of file TRAVELCCM_Service.cpp.

24.11.2.2   TRAVELCCM::TRAVELCCM_Service::TRAVELCCM_Service (const stdair::BasLogParams & *iLogParams*)

Constructor.

The init() method is called; see the corresponding documentation for more details.

A reference on an output stream is given, so that log outputs can be directed onto that

stream.

Parameters

const	stdair::BasLogParams& Parameters for the output log stream.

Definition at line 57 of file TRAVELCCM_Service.cpp.

24.11.2.3 TRAVELCCM::TRAVELCCM_Service::TRAVELCCM_Service (stdair::STDAIR_ServicePtr_T *ioSTDAIR_Service_ptr*)

Constructor.

The init() method is called; see the corresponding documentation for more details.

Moreover, as no reference on any output stream is given, it is assumed that the StdAir log service has already been initialised with the proper log output stream by some other methods in the calling chain (for instance, when the TRAVELCCM_Service is itself being initialised by another library service such as DSIM_Service).

Parameters

stdair::STDAIR_ServicePtr_-T	Reference on the STDAIR service.

Definition at line 78 of file TRAVELCCM_Service.cpp.

24.11.2.4 TRAVELCCM::TRAVELCCM_Service::∼TRAVELCCM_Service ( )

Destructor.

Definition at line 94 of file TRAVELCCM_Service.cpp.

24.11.3 Member Function Documentation

24.11.3.1 void TRAVELCCM::TRAVELCCM_Service::buildSampleBom ( )

Build a sample BOM tree, and attach it to the BomRoot instance.

Definition at line 170 of file TRAVELCCM_Service.cpp.

24.11.3.2 void TRAVELCCM::TRAVELCCM_Service::buildSampleTravelSolutions (stdair::TravelSolutionList_T & *ioTSList*)

Build a sample list of travel solutions.

As of now (March 2011), that list is made of the following travel solutions:

- BA9

- LHR-SYD

- 2011-06-10

- Q

- WTP: 900

- Change fee: 20; Non refundable; Saturday night stay

Parameters

stdair::TravelS T&	Sample list of travel solution structures. It should be given empty. It is altered with the returned sample.

Definition at line 217 of file TRAVELCCM_Service.cpp.

24.11.3.3   stdair::BookingRequestStruct TRAVELCCM::TRAVELCCM_-Service::buildSampleBookingRequest (const bool *isForCRS =* false)

Build a sample booking request structure.

As of now (March 2011), the sample booking request is made of the following parameters:

- Return trip (inbound): LHR-SYD (POS: LHR, Channel: DN),

- Departing 10-JUN-2011 around 8:00, staying 7 days

- Requested on 15-MAY-2011 at 10:00

- Economy cabin, 3 persons, FF member

- WTP: 1000.0 EUR

- Dis-utility: 100.0 EUR/hour

As of now (March 2011), the CRS-related booking request is made of the following parameters:

- Return trip (inbound): SIN-BKK (POS: SIN, Channel: IN),

- Departing 30-JAN-2010 around 10:00, staying 7 days

- Requested on 22-JAN-2010 at 10:00

- Economy cabin, 3 persons, FF member

- WTP: 1000.0 EUR

- Dis-utility: 100.0 EUR/hour

Parameters

const	bool isForCRS Whether the sample booking request is for CRS.

Returns

   BookingRequestStruct& Sample booking request structure.

Definition at line 239 of file TRAVELCCM_Service.cpp.

24.11.3.4   const stdair::TravelSolutionStruct ∗ TRAVELCCM::TRAVELCCM-
         Service::chooseTravelSolution (stdair::TravelSolutionList_T & *ioTravelSolutionList,*
         const stdair::BookingRequestStruct & *iBookingRequest* **)**

Choose the travel solution and the fare option within the given list of travel solutions.

The returned pointer will be NULL if no travel solution is chosen (e.g., when the Willingness-To-Pay is too low).

Parameters

	The list of travel solution to choose among.
stdair::TravelS *T&*	
const	stdair::BookingRequestStruct& The booking request and its context.

Returns

   stdair::TravelSolutionStruct∗ The chosen travel solution. NULL when not found.

Definition at line 303 of file TRAVELCCM_Service.cpp.

24.11.3.5   std::string TRAVELCCM::TRAVELCCM_Service::csvDisplay ( ) const

Recursively display (dump in the returned string) the objects of the BOM tree.

Returns

   std::string Output string in which the BOM tree is logged/dumped.

Definition at line 260 of file TRAVELCCM_Service.cpp.

24.11.3.6   std::string TRAVELCCM::TRAVELCCM_Service::csvDisplay (const
         stdair::TravelSolutionList_T & *iTravelSolutionList* **) const**

Display (dump in the returned string) the full list of travel solution structures.

Returns

   std::string Output string in which the list of travel solutions is logged/dumped.

Definition at line 282 of file TRAVELCCM_Service.cpp.

The documentation for this class was generated from the following files:

- travelccm/TRAVELCCM_Service.hpp
- travelccm/service/TRAVELCCM_Service.cpp

24.12   TRAVELCCM::TRAVELCCM_ServiceContext Class Reference

Inner class holding the context for the TravelCCM Service object.

```
#include <travelccm/service/TRAVELCCM_ServiceContext.hpp>
```

Inheritance diagram for TRAVELCCM::TRAVELCCM_ServiceContext:
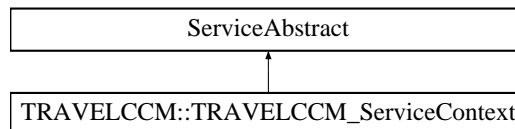
Friends

- class TRAVELCCM_Service
- class FacTRAVELCCMServiceContext

24.12.1   Detailed Description

Inner class holding the context for the TravelCCM Service object.

Definition at line 23 of file TRAVELCCM_ServiceContext.hpp.

24.12.2   Friends And Related Function Documentation

24.12.2.1   friend class **TRAVELCCM_Service** `[friend]`

The TRAVELCCM_Service class should be the sole class to get access to ServiceContext content: general users do not want to bother with a context interface.

Definition at line 29 of file TRAVELCCM_ServiceContext.hpp.

24.12.2.2   friend class **FacTRAVELCCMServiceContext** `[friend]`

Definition at line 30 of file TRAVELCCM_ServiceContext.hpp.

The documentation for this class was generated from the following files:

- travelccm/service/TRAVELCCM_ServiceContext.hpp
- travelccm/service/TRAVELCCM_ServiceContext.cpp

24.13   TravelChoiceTestSuite Class Reference

```
#include <test/travelccm/TravelChoiceTestSuite.hpp>
```

Inheritance diagram for TravelChoiceTestSuite:

```
                              ┌─────────────────────┐
                              │     TestFixture     │
                              └─────────────────────┘
                                        ▲
                              ┌─────────────────────┐
                              │ TravelChoiceTestSuite│
                              └─────────────────────┘
```

Public Member Functions

- void testTravelChoice ()
- TravelChoiceTestSuite ()

Protected Attributes

- std::stringstream _describeKey

24.13.1 Detailed Description

Definition at line 6 of file TravelChoiceTestSuite.hpp.

24.13.2 Constructor & Destructor Documentation

24.13.2.1 TravelChoiceTestSuite::TravelChoiceTestSuite ( )

Test some error detection functionalities. Constructor.

24.13.3 Member Function Documentation

24.13.3.1 void TravelChoiceTestSuite::testTravelChoice ( )

Test the Optimisation functionality.

24.13.4 Member Data Documentation

24.13.4.1 std::stringstream TravelChoiceTestSuite::_describeKey `[protected]`

Definition at line 23 of file TravelChoiceTestSuite.hpp.

The documentation for this class was generated from the following file:

- test/travelccm/TravelChoiceTestSuite.hpp

25 File Documentation

25.1 doc/local/authors.doc File Reference

25.2 doc/local/codingrules.doc File Reference

25.3 doc/local/copyright.doc File Reference

25.4 doc/local/documentation.doc File Reference

25.5 doc/local/features.doc File Reference

25.6 doc/local/help_wanted.doc File Reference

25.7 doc/local/howto_release.doc File Reference

25.8 doc/local/index.doc File Reference

25.9 doc/local/installation.doc File Reference

25.10 doc/local/linking.doc File Reference

25.11 doc/local/test.doc File Reference

25.12 doc/local/users_guide.doc File Reference

25.13 doc/local/verification.doc File Reference

25.14 doc/tutorial/tutorial.doc File Reference

25.15 test/travelccm/TravelChoiceTestSuite.cpp File Reference

25.16 TravelChoiceTestSuite.cpp

```
00001
00005 // //////////////////////////////////////////////////////////////////
00006 // Import section
00007 // //////////////////////////////////////////////////////////////////
00008 // STL
00009 #include <sstream>
00010 #include <fstream>
00011 #include <string>
00012 // Boost Unit Test Framework (UTF)
00013 #define BOOST_TEST_DYN_LINK
00014 #define BOOST_TEST_MAIN
00015 #define BOOST_TEST_MODULE TravelCCMTest
00016 #include <boost/test/unit_test.hpp>
00017 // StdAir
00018 #include <stdair/basic/BasLogParams.hpp>
00019 #include <stdair/basic/BasDBParams.hpp>
00020 #include <stdair/basic/BasFileMgr.hpp>
00021 #include <stdair/bom/TravelSolutionStruct.hpp>
```

```
00022 #include <stdair/bom/BookingRequestStruct.hpp>
00023 #include <stdair/service/Logger.hpp>
00024 // TravelCCM
00025 #include <travelccm/TRAVELCCM_Service.hpp>
00026 #include <travelccm/config/travelccm-paths.hpp>
00027
00028 namespace boost_utf = boost::unit_test;
00029
00030 // (Boost) Unit Test XML Report
00031 std::ofstream utfReportStream ("TravelChoiceTestSuite_utfresults.xml");
00032
00036 struct UnitTestConfig {
00038   UnitTestConfig() {
00039     boost_utf::unit_test_log.set_stream (utfReportStream);
00040     boost_utf::unit_test_log.set_format (boost_utf::XML);
00041     boost_utf::unit_test_log.set_threshold_level (boost_utf::log_test_units);
00042     //boost_utf::unit_test_log.set_threshold_level (boost_utf::log_successful_tes
    ts);
00043   }
00045   ~UnitTestConfig() {
00046   }
00047 };
00048
00049
00050 // /////////////// Main: Unit Test Suite ///////////////
00051
00052 // Set the UTF configuration (re-direct the output to a specific file)
00053 BOOST_GLOBAL_FIXTURE (UnitTestConfig);
00054
00055 // Start the test suite
00056 BOOST_AUTO_TEST_SUITE (master_test_suite)
00057
00058
00061 BOOST_AUTO_TEST_CASE (simple_simulation_test) {
00062
00063   // Input file name
00064   /*
00065     const stdair::Filename_T inputFileName (STDAIR_SAMPLE_DIR "/ccm_02.csv");
00066
00067   // Check that the file path given as input corresponds to an actual file
00068   const bool doesExistAndIsReadable =
00069     stdair::BasFileMgr::doesExistAndIsReadable (lInputFilename);
00070   BOOST_CHECK_MESSAGE (doesExistAndIsReadable == true,
00071                        "The '" << lInputFilename
00072                        << "' input file can not be open and read");
00073   */
00074
00075   // Output log File
00076   const stdair::Filename_T lLogFilename ("TravelChoiceTestSuite.log");
00077
00078   // Set the log parameters
00079   std::ofstream logOutputFile;
00080   // Open and clean the log outputfile
00081   logOutputFile.open (lLogFilename.c_str());
00082   logOutputFile.clear();
00083
00084   // Initialise the service context
00085   const stdair::BasLogParams lLogParams (stdair::LOG::DEBUG, logOutputFile);
00086
00087   // Build the BOM tree
00088   TRAVELCCM::TRAVELCCM_Service travelccmService (lLogParams);
00089
```

```
00090   // DEBUG
00091   STDAIR_LOG_DEBUG ("Welcome to TravelCCM");
00092
00093   // Build a list of travel solutions
00094   const stdair::BookingRequestStruct& lBookingRequest =
00095     travelccmService.buildSampleBookingRequest();
00096
00097   // DEBUG
00098   STDAIR_LOG_DEBUG ("Booking request: " << lBookingRequest.display());
00099
00100   // Build the sample BOM tree
00101   stdair::TravelSolutionList_T lTSList;
00102   travelccmService.buildSampleTravelSolutions (lTSList);
00103
00104   // DEBUG: Display the list of travel solutions
00105   const std::string& lCSVDump = travelccmService.csvDisplay (lTSList);
00106   STDAIR_LOG_DEBUG (lCSVDump);
00107
00108   // Choose a travel solution
00109   const stdair::TravelSolutionStruct* lTS_ptr =
00110     travelccmService.chooseTravelSolution (lTSList, lBookingRequest);
00111
00112   // Check that a solution has been found
00113   BOOST_REQUIRE_MESSAGE (lTS_ptr != NULL,
00114                          "No travel solution can be found for "
00115                          << lBookingRequest.display()
00116                          << " within the following list of travel solutions. "
00117                          << lCSVDump);
00118
00119   // Close the log file
00120   logOutputFile.close();
00121 }
00122
00123 // End the test suite
00124 BOOST_AUTO_TEST_SUITE_END()
00125
00126
```

25.17   test/travelccm/TravelChoiceTestSuite.hpp File Reference

`#include <sstream>`

`#include <cppunit/extensions/HelperMacros.h>`

Classes

- class TravelChoiceTestSuite

Functions

- CPPUNIT_TEST_SUITE_REGISTRATION (TravelChoiceTestSuite)

25.17.1   Function Documentation

25.17.1.1 CPPUNIT_TEST_SUITE_REGISTRATION (TravelChoiceTestSuite)

25.18 TravelChoiceTestSuite.hpp

```
00001 // STL
00002 #include <sstream>
00003 // CPPUNIT
00004 #include <cppunit/extensions/HelperMacros.h>
00005
00006 class TravelChoiceTestSuite : public CppUnit::TestFixture {
00007   CPPUNIT_TEST_SUITE (TravelChoiceTestSuite);
00008   CPPUNIT_TEST (testTravelChoice);
00009   //  CPPUNIT_TEST (errorCase);
00010   CPPUNIT_TEST_SUITE_END ();
00011 public:
00012
00014   void testTravelChoice ();
00015
00017   //  void errorCase ();
00018
00020   TravelChoiceTestSuite ();
00021
00022 protected:
00023   std::stringstream _describeKey;
00024 };
00025 CPPUNIT_TEST_SUITE_REGISTRATION (TravelChoiceTestSuite);
```

25.19 travelccm/basic/BasConst.cpp File Reference

```
#include <travelccm/basic/BasConst_General.hpp>
```

Namespaces

- namespace TRAVELCCM

25.20 BasConst.cpp

```
00001 // //////////////////////////////////////////////////////////////////////
00002 // Import section
00003 // //////////////////////////////////////////////////////////////////////
00004 #include <travelccm/basic/BasConst_General.hpp>
00005
00006 namespace TRAVELCCM {
00007
00008 }
```

25.21 travelccm/basic/BasConst_General.hpp File Reference

Namespaces

- namespace TRAVELCCM

25.22   BasConst␣General.hpp

```
00001 #ifndef __TRAVELCCM_BAS_BASCONST_GENERAL_HPP
00002 #define __TRAVELCCM_BAS_BASCONST_GENERAL_HPP
00003
00004 // //////////////////////////////////////////////////////////////////////
00005 // Import section
00006 // //////////////////////////////////////////////////////////////////////
00007
00008 namespace TRAVELCCM {
00009
00010 }
00011 #endif // __TRAVELCCM_BAS_BASCONST_GENERAL_HPP
```

25.23   travelccm/batches/travelccm.cpp File Reference

25.24   travelccm.cpp

```
00001
00005 // STL
00006 #include <cassert>
00007 #include <iostream>
00008 #include <sstream>
00009 #include <fstream>
00010 #include <string>
00011 // Boost (Extended STL)
00012 #include <boost/program_options.hpp>
00013 // StdAir
00014 #include <stdair/basic/BasLogParams.hpp>
00015 #include <stdair/basic/BasDBParams.hpp>
00016 #include <stdair/bom/BookingRequestStruct.hpp>
00017 #include <stdair/bom/TravelSolutionStruct.hpp>
00018 #include <stdair/service/Logger.hpp>
00019 // TravelCCM
00020 #include <travelccm/TRAVELCCM_Service.hpp>
00021 #include <travelccm/config/travelccm-paths.hpp>
00022
00023 // //////// Constants //////
00027 const std::string K_TRAVELCCM_DEFAULT_LOG_FILENAME ("travelccm.log");
00028
00032 const std::string K_TRAVELCCM_DEFAULT_INPUT_FILENAME (STDAIR_SAMPLE_DIR
00033                                                       "/ccm_01.csv");
00034
00039 const bool K_TRAVELCCM_DEFAULT_BUILT_IN_INPUT = false;
00040
00044 const int K_TRAVELCCM_EARLY_RETURN_STATUS = 99;
00045
00046
00047 // ///////// Parsing of Options & Configuration /////////
00048 // A helper function to simplify the main part.
00049 template<class T> std::ostream& operator<< (std::ostream& os,
00050                                             const std::vector<T>& v) {
00051   std::copy (v.begin(), v.end(), std::ostream_iterator<T> (std::cout, " "));
00052   return os;
00053 }
00054
00058 int readConfiguration (int argc, char* argv[], bool& ioIsBuiltin,
00059                        stdair::Filename_T& lInputFilename,
00060                        stdair::Filename_T& lLogFilename) {
```

```
00061
00062   // Default for the built-in input
00063   ioIsBuiltin = K_TRAVELCCM_DEFAULT_BUILT_IN_INPUT;
00064
00065   // Declare a group of options that will be allowed only on command line
00066   boost::program_options::options_description generic ("Generic options");
00067   generic.add_options()
00068     ("prefix", "print installation prefix")
00069     ("version,v", "print version string")
00070     ("help,h", "produce help message");
00071
00072   // Declare a group of options that will be allowed both on command
00073   // line and in config file
00074   boost::program_options::options_description config ("Configuration");
00075   config.add_options()
00076     ("builtin,b",
00077      "The sample BOM tree can be either built-in or parsed from an input file. Th
     at latter must then be given with the -i/--input option")
00078     ("input,i",
00079      boost::program_options::value< std::string >(&lInputFilename)->default_value
     (K_TRAVELCCM_DEFAULT_INPUT_FILENAME),
00080      "(CSV) input file for the customer choice rule sets")
00081     ("log,l",
00082      boost::program_options::value< std::string >(&lLogFilename)->default_value(K
     _TRAVELCCM_DEFAULT_LOG_FILENAME),
00083      "Filename for the logs")
00084     ;
00085
00086   // Hidden options, will be allowed both on command line and
00087   // in config file, but will not be shown to the user.
00088   boost::program_options::options_description hidden ("Hidden options");
00089   hidden.add_options()
00090     ("copyright",
00091      boost::program_options::value< std::vector<std::string> >(),
00092      "Show the copyright (license)");
00093
00094   boost::program_options::options_description cmdline_options;
00095   cmdline_options.add(generic).add(config).add(hidden);
00096
00097   boost::program_options::options_description config_file_options;
00098   config_file_options.add(config).add(hidden);
00099
00100   boost::program_options::options_description visible ("Allowed options");
00101   visible.add(generic).add(config);
00102
00103   boost::program_options::positional_options_description p;
00104   p.add ("copyright", -1);
00105
00106   boost::program_options::variables_map vm;
00107   boost::program_options::
00108     store (boost::program_options::command_line_parser (argc, argv).
00109            options (cmdline_options).positional(p).run(), vm);
00110
00111   std::ifstream ifs ("travelccm.cfg");
00112   boost::program_options::store (parse_config_file (ifs, config_file_options),
00113                                  vm);
00114   boost::program_options::notify (vm);
00115
00116   if (vm.count ("help")) {
00117     std::cout << visible << std::endl;
00118     return K_TRAVELCCM_EARLY_RETURN_STATUS;
00119   }
```

```
00120
00121   if (vm.count ("version")) {
00122     std::cout << PACKAGE_NAME << ", version " << PACKAGE_VERSION << std::endl;
00123     return K_TRAVELCCM_EARLY_RETURN_STATUS;
00124   }
00125
00126   if (vm.count ("prefix")) {
00127     std::cout << "Installation prefix: " << PREFIXDIR << std::endl;
00128     return K_TRAVELCCM_EARLY_RETURN_STATUS;
00129   }
00130
00131   if (vm.count ("builtin")) {
00132     ioIsBuiltin = true;
00133   }
00134   const std::string isBuiltinStr = (ioIsBuiltin == true)?"yes":"no";
00135   std::cout << "The BOM should be built-in? " << isBuiltinStr << std::endl;
00136
00137   if (ioIsBuiltin == false) {
00138
00139     // The BOM tree should be built from parsing a customer-choice rule file
00140     if (vm.count ("input")) {
00141       lInputFilename = vm["input"].as< std::string >();
00142       std::cout << "Input filename is: " << lInputFilename << std::endl;
00143
00144     } else {
00145       // The built-in option is not selected. However, no demand input file
00146       // is specified
00147       std::cerr << "Either one among the -b/--builtin and -i/--input "
00148                 << "options must be specified" << std::endl;
00149     }
00150   }
00151
00152   if (vm.count ("log")) {
00153     lLogFilename = vm["log"].as< std::string >();
00154     std::cout << "Log filename is: " << lLogFilename << std::endl;
00155   }
00156
00157   return 0;
00158 }
00159
00160
00161 // ///////// M A I N ////////////
00162 int main (int argc, char* argv[]) {
00163
00164   // State whether the BOM tree should be built-in or parsed from an input file
00165   bool isBuiltin;
00166
00167   // Input file name
00168   stdair::Filename_T lInputFilename;
00169
00170   // Output log File
00171   stdair::Filename_T lLogFilename;
00172
00173   // Call the command-line option parser
00174   const int lOptionParserStatus =
00175     readConfiguration (argc, argv, isBuiltin, lInputFilename, lLogFilename);
00176
00177   if (lOptionParserStatus == K_TRAVELCCM_EARLY_RETURN_STATUS) {
00178     return 0;
00179   }
00180
00181   // Set the log parameters
```

```
00182    std::ofstream logOutputFile;
00183    // Open and clean the log outputfile
00184    logOutputFile.open (lLogFilename.c_str());
00185    logOutputFile.clear();
00186
00187    // Initialise the service context
00188    const stdair::BasLogParams lLogParams (stdair::LOG::DEBUG, logOutputFile);
00189
00190    // Build the BOM tree
00191    TRAVELCCM::TRAVELCCM_Service travelccmService (lLogParams);
00192
00193    // DEBUG
00194    STDAIR_LOG_DEBUG ("Welcome to TravelCCM");
00195
00196    // Check wether or not a (CSV) input file should be read
00197    if (isBuiltin == true) {
00198      // Create a sample Customer-Choice rule object, and insert it
00199      // within the BOM tree
00200      travelccmService.buildSampleBom();
00201
00202    } else {
00209      // travelccmService.parseAndLoad (lInputFilename);
00210    }
00211
00212    // Build a list of travel solutions
00213    const stdair::BookingRequestStruct& lBookingRequest =
00214      travelccmService.buildSampleBookingRequest();
00215
00216    // DEBUG
00217    STDAIR_LOG_DEBUG ("Booking request: " << lBookingRequest.display());
00218
00219    // Build the sample BOM tree
00220    stdair::TravelSolutionList_T lTSList;
00221    travelccmService.buildSampleTravelSolutions (lTSList);
00222
00223    // DEBUG: Display the list of travel solutions
00224    const std::string& lCSVDump = travelccmService.csvDisplay (lTSList);
00225    STDAIR_LOG_DEBUG (lCSVDump);
00226
00227    // Choose a travel solution
00228    const stdair::TravelSolutionStruct* lTS_ptr =
00229      travelccmService.chooseTravelSolution (lTSList, lBookingRequest);
00230
00231    if (lTS_ptr != NULL) {
00232      // DEBUG
00233      STDAIR_LOG_DEBUG ("Chosen travel solution: " << lTS_ptr->display());
00234
00235    } else {
00236      // DEBUG
00237      STDAIR_LOG_DEBUG ("No travel solution can be found for "
00238                        << lBookingRequest.display()
00239                        << " within the following list of travel solutions");
00240      STDAIR_LOG_DEBUG (lCSVDump);
00241    }
00242
00243    // Close the Log outputFile
00244    logOutputFile.close();
00245
00253    return 0;
00254  }
```

25.25   travelccm/bom/PriceOrientedModel.cpp File Reference

```
#include <cassert>

#include <sstream>

#include <stdair/bom/BomKeyManager.hpp>

#include <stdair/bom/BookingClassKey.hpp>

#include <stdair/bom/BookingRequestStruct.hpp>

#include <stdair/bom/TravelSolutionStruct.hpp>

#include <stdair/bom/FareOptionStruct.hpp>

#include <stdair/service/Logger.hpp>

#include <travelccm/bom/PriceOrientedModel.hpp>
```

Namespaces

- namespace TRAVELCCM

25.26   PriceOrientedModel.cpp

```
00001 // //////////////////////////////////////////////////////////////////////
00002 // Import section
00003 // //////////////////////////////////////////////////////////////////////
00004 // STL
00005 #include <cassert>
00006 #include <sstream>
00007 // StdAir
00008 #include <stdair/bom/BomKeyManager.hpp>
00009 #include <stdair/bom/BookingClassKey.hpp>
00010 #include <stdair/bom/BookingRequestStruct.hpp>
00011 #include <stdair/bom/TravelSolutionStruct.hpp>
00012 #include <stdair/bom/FareOptionStruct.hpp>
00013 #include <stdair/service/Logger.hpp>
00014 // TravelCCM
00015 #include <travelccm/bom/PriceOrientedModel.hpp>
00016
00017 namespace TRAVELCCM {
00018
00019   // //////////////////////////////////////////////////////////////////////
00020   const stdair::TravelSolutionStruct* PriceOrientedModel::
00021   chooseTravelSolution (stdair::TravelSolutionList_T& ioTSList,
00022                         const stdair::BookingRequestStruct& iBookingRequest) {
00023     stdair::TravelSolutionStruct* oChosenTS_ptr = NULL;
00024
00025     // Retrieve the number of passengers
00026     const stdair::NbOfSeats_T& lPartySize = iBookingRequest.getPartySize();
00027
00028     // Retrieve the Willingness-to-Pay (WTP) of the customer
00029     const stdair::WTP_T& lWTP = iBookingRequest.getWTP();
00030
00031     // Browse the travel solution list and choose the cheapest one
00032     stdair::Fare_T lLowestFare = std::numeric_limits<stdair::Fare_T>::max();
00033     for (stdair::TravelSolutionList_T::iterator itTS = ioTSList.begin();
```

```
00034             itTS != ioTSList.end(); ++itTS) {
00035        stdair::TravelSolutionStruct& lTS = *itTS;
00036
00037        // Browse the fare options
00038        const stdair::FareOptionList_T& lFOList = lTS.getFareOptionList();
00039        for (stdair::FareOptionList_T::const_iterator itFO = lFOList.begin();
00040             itFO != lFOList.end(); ++itFO) {
00041          const stdair::FareOptionStruct& lFO = *itFO;
00042
00043          // Choose the current fare option and the current solution
00044          // if the current fare is lower than the current lowest fare.
00045          const stdair::Fare_T& lFOFare = lFO.getFare();
00046          const stdair::Availability_T& lFOAvl = lFO.getAvailability();
00047
00048          if (lFOFare < lLowestFare && lFOFare <= lWTP && lFOAvl >= lPartySize) {
00049
00050            // DEBUG
00051            /*
00052            STDAIR_LOG_DEBUG ("The travel solution (TS) '" << lTS
00053                              << "' is chosen because its fare (" << lFOFare
00054                              << ") is lower than the lowest fare (" << lLowestFare

00055                              << ") and than the WTP (" << lWTP
00056                              << "), and because the party size (" << lPartySize
00057                              << ") is lower than the availability (" << lFOAvl
00058                              << ")");
00059            */
00060
00061            lLowestFare = lFOFare;
00062            oChosenTS_ptr = &lTS;
00063            oChosenTS_ptr->setChosenFareOption (lFO);
00064
00065          } else {
00066            // DEBUG
00067            /*
00068            STDAIR_LOG_DEBUG ("The travel solution (TS) '" << lTS
00069                              << "' is not chosen because either its fare ("
00070                              << lFOFare << ") is greater than the lowest fare ("
00071                              << lLowestFare << ") or than the WTP (" << lWTP
00072                              << "), or because the party size (" << lPartySize
00073                              << ") is greater than the availability (" << lFOAvl
00074                              << ")");
00075            */
00076          }
00077        }
00078      }
00079
00080      return oChosenTS_ptr;
00081    }
00082
00083 }
```

25.27 travelccm/bom/PriceOrientedModel.hpp File Reference

```
#include <stdair/stdair_service_types.hpp>

#include <stdair/bom/TravelSolutionTypes.hpp>
```

Classes

- class TRAVELCCM::PriceOrientedModel

Namespaces

- namespace stdair

  Forward declarations.
- namespace TRAVELCCM

25.28   PriceOrientedModel.hpp

```
00001 #ifndef __TRAVELCCM_BOM_PRICEORIENTEDMODEL_HPP
00002 #define __TRAVELCCM_BOM_PRICEORIENTEDMODEL_HPP
00003
00004 // //////////////////////////////////////////////////////////////////////
00005 // Import section
00006 // //////////////////////////////////////////////////////////////////////
00007 // StdAir
00008 #include <stdair/stdair_service_types.hpp>
00009 #include <stdair/bom/TravelSolutionTypes.hpp>
00010
00012 namespace stdair {
00013   struct BookingRequestStruct;
00014 }
00015
00016 namespace TRAVELCCM {
00017
00021   class PriceOrientedModel {
00022   public:
00030     static const stdair::TravelSolutionStruct*
00031     chooseTravelSolution (stdair::TravelSolutionList_T&,
00032                           const stdair::BookingRequestStruct&);
00033   };
00034 }
00035 #endif // __TRAVELCCM_BOM_PRICEORIENTEDMODEL_HPP
```

25.29   travelccm/command/ChoiceManager.cpp File Reference

#include <cassert>

#include <stdair/bom/TravelSolutionStruct.hpp>

#include <stdair/bom/BookingRequestStruct.hpp>

#include <travelccm/bom/PriceOrientedModel.hpp>

#include <travelccm/command/ChoiceManager.hpp>

Namespaces

- namespace TRAVELCCM

25.30 ChoiceManager.cpp

```
00001 // //////////////////////////////////////////////////////////////////////
00002 // Import section
00003 // //////////////////////////////////////////////////////////////////////
00004 // STL
00005 #include <cassert>
00006 // StdAir
00007 #include <stdair/bom/TravelSolutionStruct.hpp>
00008 #include <stdair/bom/BookingRequestStruct.hpp>
00009 // TravelCCM
00010 #include <travelccm/bom/PriceOrientedModel.hpp>
00011 #include <travelccm/command/ChoiceManager.hpp>
00012
00013 namespace TRAVELCCM {
00014
00015   // //////////////////////////////////////////////////////////////////////
00016   const stdair::TravelSolutionStruct* ChoiceManager::
00017   chooseTravelSolution (stdair::TravelSolutionList_T& ioTravelSolutionList,
00018                         const stdair::BookingRequestStruct& iBookingRequest) {
00019     // Return the cheapest solution.
00020     const stdair::TravelSolutionStruct* oTravelSolution_ptr =
00021       PriceOrientedModel::chooseTravelSolution (ioTravelSolutionList,
00022                                                 iBookingRequest);
00023     return oTravelSolution_ptr;
00024   }
00025
00026 }
```

25.31 travelccm/command/ChoiceManager.hpp File Reference

```
#include <stdair/stdair_service_types.hpp>
```

```
#include <stdair/bom/TravelSolutionTypes.hpp>
```

Classes

- class TRAVELCCM::ChoiceManager

Namespaces

- namespace stdair

    Forward declarations.

- namespace TRAVELCCM

25.32 ChoiceManager.hpp

```
00001 #ifndef __TRAVELCCM_CMD_CHOICEMANAGER_HPP
00002 #define __TRAVELCCM_CMD_CHOICEMANAGER_HPP
00003
00004 // //////////////////////////////////////////////////////////////////////
00005 // Import section
00006 // //////////////////////////////////////////////////////////////////////
```

```
00007 // StdAir
00008 #include <stdair/stdair_service_types.hpp>
00009 #include <stdair/bom/TravelSolutionTypes.hpp>
00010
00012 namespace stdair {
00013   struct BookingRequestStruct;
00014 }
00015
00016 namespace TRAVELCCM {
00017
00021   class ChoiceManager {
00022   public:
00030     static const stdair::TravelSolutionStruct*
00031     chooseTravelSolution (stdair::TravelSolutionList_T&,
00032                           const stdair::BookingRequestStruct&);
00033   };
00034 }
00035 #endif // __TRAVELCCM_CMD_CHOICEMANAGER_HPP
```

25.33   travelccm/command/CmdAbstract.cpp File Reference

```
#include <travelccm/command/CmdAbstract.hpp>
```

Namespaces

- namespace TRAVELCCM

25.34   CmdAbstract.cpp

```
00001 // //////////////////////////////////////////////////////////////////////
00002 // Import section
00003 // //////////////////////////////////////////////////////////////////////
00004 // TRAVELCCM
00005 #include <travelccm/command/CmdAbstract.hpp>
00006
00007 namespace TRAVELCCM {
00008
00009 }
```

25.35   travelccm/command/CmdAbstract.hpp File Reference

Classes

- class TRAVELCCM::CmdAbstract

Namespaces

- namespace TRAVELCCM

25.36   CmdAbstract.hpp

```
00001 #ifndef __TRAVELCCM_CMD_CMDABSTRACT_HPP
00002 #define __TRAVELCCM_CMD_CMDABSTRACT_HPP
00003
00004 // //////////////////////////////////////////////////////////////////////
00005 // Import section
00006 // //////////////////////////////////////////////////////////////////////
00007
00008 namespace TRAVELCCM {
00009
00011   class CmdAbstract {
00012   public:
00013
00014   };
00015
00016 }
00017 #endif // __TRAVELCCM_CMD_CMDABSTRACT_HPP
```

25.37   travelccm/command/FileMgr.cpp File Reference

```
#include <cassert>
```

```
#include <fstream>
```

```
#include <stdair/service/Logger.hpp>
```

```
#include <travelccm/command/FileMgr.hpp>
```

Namespaces

- namespace TRAVELCCM

25.38   FileMgr.cpp

```
00001 // //////////////////////////////////////////////////////////////////////
00002 // Import section
00003 // //////////////////////////////////////////////////////////////////////
00004 // STL
00005 #include <cassert>
00006 #include <fstream>
00007 // StdAir
00008 #include <stdair/service/Logger.hpp>
00009 // TRAVEL-CCM
00010 #include <travelccm/command/FileMgr.hpp>
00011
00012 namespace TRAVELCCM {
00013
00014 }
```

25.39   travelccm/command/FileMgr.hpp File Reference

```
#include <string>
```

Classes

- class TRAVELCCM::FileMgr

Namespaces

- namespace TRAVELCCM

25.40   FileMgr.hpp

```
00001 #ifndef __TRAVELCCM_CMD_FILEMGR_HPP
00002 #define __TRAVELCCM_CMD_FILEMGR_HPP
00003
00004 // //////////////////////////////////////////////////////////////////////
00005 // Import section
00006 // //////////////////////////////////////////////////////////////////////
00007 #include <string>
00008
00009 namespace TRAVELCCM {
00010
00011   class TravelSolutionHolder;
00012
00015   class FileMgr {
00016   public:
00017
00018   };
00019 }
00020 #endif // __TRAVELCCM_CMD_FILEMGR_HPP
```

25.41   travelccm/config/travelccm-paths.hpp File Reference

Defines

- #define PACKAGE "travelccm"
- #define PACKAGE_NAME "TRAVELCCM"
- #define PACKAGE_VERSION "0.5.3"
- #define PREFIXDIR "/usr"
- #define EXEC_PREFIX "/usr"
- #define BINDIR "/usr/bin"
- #define LIBDIR "/usr/lib64"
- #define LIBEXECDIR "/usr/libexec"
- #define SBINDIR "/usr/sbin"
- #define SYSCONFDIR "/usr/etc"
- #define INCLUDEDIR "/usr/include"
- #define DATAROOTDIR "/usr/share"
- #define DATADIR "/usr/share"
- #define DOCDIR "/usr/share/doc/travelccm-0.5.3"
- #define MANDIR "/usr/share/man"
- #define INFODIR "/usr/share/info"
- #define HTMLDIR "/usr/share/doc/travelccm-0.5.3/html"

- #define PDFDIR "/usr/share/doc/travelccm-0.5.3/html"
- #define STDAIR_SAMPLE_DIR "/usr/share/stdair/samples"

25.41.1   Define Documentation

25.41.1.1   #define PACKAGE "travelccm"

Definition at line 4 of file travelccm-paths.hpp.

25.41.1.2   #define PACKAGE_NAME "TRAVELCCM"

Definition at line 5 of file travelccm-paths.hpp.

25.41.1.3   #define PACKAGE_VERSION "0.5.3"

Definition at line 6 of file travelccm-paths.hpp.

25.41.1.4   #define PREFIXDIR "/usr"

Definition at line 7 of file travelccm-paths.hpp.

25.41.1.5   #define EXEC_PREFIX "/usr"

Definition at line 8 of file travelccm-paths.hpp.

25.41.1.6   #define BINDIR "/usr/bin"

Definition at line 9 of file travelccm-paths.hpp.

25.41.1.7   #define LIBDIR "/usr/lib64"

Definition at line 10 of file travelccm-paths.hpp.

25.41.1.8   #define LIBEXECDIR "/usr/libexec"

Definition at line 11 of file travelccm-paths.hpp.

25.41.1.9   #define SBINDIR "/usr/sbin"

Definition at line 12 of file travelccm-paths.hpp.

25.41.1.10   #define SYSCONFDIR "/usr/etc"

Definition at line 13 of file travelccm-paths.hpp.

25.41.1.11   #define INCLUDEDIR "/usr/include"

Definition at line 14 of file travelccm-paths.hpp.

25.41.1.12   #define DATAROOTDIR "/usr/share"

Definition at line 15 of file travelccm-paths.hpp.

25.41.1.13   #define DATADIR "/usr/share"

Definition at line 16 of file travelccm-paths.hpp.

25.41.1.14   #define DOCDIR "/usr/share/doc/travelccm-0.5.3"

Definition at line 17 of file travelccm-paths.hpp.

25.41.1.15   #define MANDIR "/usr/share/man"

Definition at line 18 of file travelccm-paths.hpp.

25.41.1.16   #define INFODIR "/usr/share/info"

Definition at line 19 of file travelccm-paths.hpp.

25.41.1.17   #define HTMLDIR "/usr/share/doc/travelccm-0.5.3/html"

Definition at line 20 of file travelccm-paths.hpp.

25.41.1.18   #define PDFDIR "/usr/share/doc/travelccm-0.5.3/html"

Definition at line 21 of file travelccm-paths.hpp.

25.41.1.19   #define STDAIR_SAMPLE_DIR "/usr/share/stdair/samples"

Definition at line 22 of file travelccm-paths.hpp.

25.42   travelccm-paths.hpp

```
00001 #ifndef __TRAVELCCM_PATHS_HPP__
00002 #define __TRAVELCCM_PATHS_HPP__
00003
00004 #define PACKAGE "travelccm"
00005 #define PACKAGE_NAME "TRAVELCCM"
00006 #define PACKAGE_VERSION "0.5.3"
00007 #define PREFIXDIR "/usr"
00008 #define EXEC_PREFIX "/usr"
00009 #define BINDIR "/usr/bin"
00010 #define LIBDIR "/usr/lib64"
00011 #define LIBEXECDIR "/usr/libexec"
00012 #define SBINDIR "/usr/sbin"
00013 #define SYSCONFDIR "/usr/etc"
00014 #define INCLUDEDIR "/usr/include"
00015 #define DATAROOTDIR "/usr/share"
00016 #define DATADIR "/usr/share"
00017 #define DOCDIR "/usr/share/doc/travelccm-0.5.3"
00018 #define MANDIR "/usr/share/man"
00019 #define INFODIR "/usr/share/info"
00020 #define HTMLDIR "/usr/share/doc/travelccm-0.5.3/html"
00021 #define PDFDIR "/usr/share/doc/travelccm-0.5.3/html"
00022 #define STDAIR_SAMPLE_DIR "/usr/share/stdair/samples"
00023
00024 #endif // __TRAVELCCM_PATHS_HPP__
```

25.43   travelccm/config/travelccm-paths.hpp.in File Reference

Defines

- #define __TRAVELCCM_PATHS_HPP__
- #define PACKAGE "@PACKAGE@"
- #define PACKAGE_NAME "@PACKAGE_NAME@"
- #define PACKAGE_VERSION "@PACKAGE_VERSION@"
- #define PREFIXDIR "@prefix@"
- #define EXEC_PREFIX "@exec_prefix@"
- #define BINDIR "@bindir@"
- #define LIBDIR "@libdir@"
- #define LIBEXECDIR "@libexecdir@"
- #define SBINDIR "@sbindir@"
- #define SYSCONFDIR "@sysconfdir@"
- #define INCLUDEDIR "@includedir@"
- #define DATAROOTDIR "@datarootdir@"
- #define DATADIR "@datadir@"
- #define DOCDIR "@docdir@"
- #define MANDIR "@mandir@"
- #define INFODIR "@infodir@"
- #define HTMLDIR "@htmldir@"
- #define PDFDIR "@pdfdir@"
- #define STDAIR_SAMPLE_DIR "@sampledir@"

25.43.1   Define Documentation

25.43.1.1   #define __TRAVELCCM_PATHS_HPP__

Definition at line 2 of file travelccm-paths.hpp.in.

25.43.1.2   #define PACKAGE "@PACKAGE@"

Definition at line 4 of file travelccm-paths.hpp.in.

25.43.1.3   #define PACKAGE_NAME "@PACKAGE_NAME@"

Definition at line 5 of file travelccm-paths.hpp.in.

25.43.1.4   #define PACKAGE_VERSION "@PACKAGE_VERSION@"

Definition at line 6 of file travelccm-paths.hpp.in.

25.43.1.5   #define PREFIXDIR "@prefix@"

Definition at line 7 of file travelccm-paths.hpp.in.

25.43.1.6   #define EXEC_PREFIX "@exec_prefix@"

Definition at line 8 of file travelccm-paths.hpp.in.

25.43.1.7   #define BINDIR "@bindir@"

Definition at line 9 of file travelccm-paths.hpp.in.

25.43.1.8   #define LIBDIR "@libdir@"

Definition at line 10 of file travelccm-paths.hpp.in.

25.43.1.9   #define LIBEXECDIR "@libexecdir@"

Definition at line 11 of file travelccm-paths.hpp.in.

25.43.1.10   #define SBINDIR "@sbindir@"

Definition at line 12 of file travelccm-paths.hpp.in.

25.43.1.11   #define SYSCONFDIR "@sysconfdir@"

Definition at line 13 of file travelccm-paths.hpp.in.

25.43.1.12   #define INCLUDEDIR "@includedir@"

Definition at line 14 of file travelccm-paths.hpp.in.

25.43.1.13   #define DATAROOTDIR "@datarootdir@"

Definition at line 15 of file travelccm-paths.hpp.in.

25.43.1.14   #define DATADIR "@datadir@"

Definition at line 16 of file travelccm-paths.hpp.in.

25.43.1.15   #define DOCDIR "@docdir@"

Definition at line 17 of file travelccm-paths.hpp.in.

25.43.1.16   #define MANDIR "@mandir@"

Definition at line 18 of file travelccm-paths.hpp.in.

25.43.1.17   #define INFODIR "@infodir@"

Definition at line 19 of file travelccm-paths.hpp.in.

25.43.1.18   #define HTMLDIR "@htmldir@"

Definition at line 20 of file travelccm-paths.hpp.in.

25.43.1.19   #define PDFDIR "@pdfdir@"

Definition at line 21 of file travelccm-paths.hpp.in.

25.43.1.20   #define STDAIR␣SAMPLE␣DIR "@sampledir@"

Definition at line 22 of file travelccm-paths.hpp.in.

25.44   travelccm-paths.hpp.in

```
00001 #ifndef __TRAVELCCM_PATHS_HPP__
00002 #define __TRAVELCCM_PATHS_HPP__
00003
00004 #define PACKAGE "@PACKAGE@"
00005 #define PACKAGE_NAME "@PACKAGE_NAME@"
00006 #define PACKAGE_VERSION "@PACKAGE_VERSION@"
00007 #define PREFIXDIR "@prefix@"
00008 #define EXEC_PREFIX "@exec_prefix@"
00009 #define BINDIR "@bindir@"
00010 #define LIBDIR "@libdir@"
00011 #define LIBEXECDIR "@libexecdir@"
00012 #define SBINDIR "@sbindir@"
00013 #define SYSCONFDIR "@sysconfdir@"
00014 #define INCLUDEDIR "@includedir@"
00015 #define DATAROOTDIR "@datarootdir@"
00016 #define DATADIR "@datadir@"
00017 #define DOCDIR "@docdir@"
00018 #define MANDIR "@mandir@"
00019 #define INFODIR "@infodir@"
00020 #define HTMLDIR "@htmldir@"
00021 #define PDFDIR "@pdfdir@"
00022 #define STDAIR_SAMPLE_DIR "@sampledir@"
00023
00024 #endif // __TRAVELCCM_PATHS_HPP__
```

25.45   travelccm/factory/FacTRAVELCCMServiceContext.cpp File Reference

```
#include <cassert>

#include <stdair/service/FacSupervisor.hpp>

#include <travelccm/factory/FacTRAVELCCMServiceContext.hpp>

#include <travelccm/service/TRAVELCCM_ServiceContext.hpp>
```

Namespaces

- namespace TRAVELCCM

25.46   FacTRAVELCCMServiceContext.cpp

```
00001 // //////////////////////////////////////////////////////////////////////
00002 // Import section
00003 // //////////////////////////////////////////////////////////////////////
00004 // STL
00005 #include <cassert>
00006 // StdAir
00007 #include <stdair/service/FacSupervisor.hpp>
```

```
00008 // TravelCCM
00009 #include <travelccm/factory/FacTRAVELCCMServiceContext.hpp>
00010 #include <travelccm/service/TRAVELCCM_ServiceContext.hpp>
00011
00012 namespace TRAVELCCM {
00013
00014   FacTRAVELCCMServiceContext* FacTRAVELCCMServiceContext::_instance = NULL;
00015
00016   // //////////////////////////////////////////////////////////////////
00017   FacTRAVELCCMServiceContext::~FacTRAVELCCMServiceContext () {
00018     _instance = NULL;
00019   }
00020
00021   // //////////////////////////////////////////////////////////////////
00022   FacTRAVELCCMServiceContext& FacTRAVELCCMServiceContext::instance () {
00023
00024     if (_instance == NULL) {
00025       _instance = new FacTRAVELCCMServiceContext();
00026       assert (_instance != NULL);
00027
00028       stdair::FacSupervisor::instance().registerServiceFactory (_instance);
00029     }
00030     return *_instance;
00031   }
00032
00033   // //////////////////////////////////////////////////////////////////
00034   TRAVELCCM_ServiceContext& FacTRAVELCCMServiceContext::create () {
00035     TRAVELCCM_ServiceContext* aServiceContext_ptr = NULL;
00036
00037     aServiceContext_ptr = new TRAVELCCM_ServiceContext ();
00038     assert (aServiceContext_ptr != NULL);
00039
00040     // The new object is added to the Bom pool
00041     _pool.push_back (aServiceContext_ptr);
00042
00043     return *aServiceContext_ptr;
00044   }
00045
00046 }
```

25.47 travelccm/factory/FacTRAVELCCMServiceContext.hpp File Reference

#include <stdair/service/FacServiceAbstract.hpp>

#include <travelccm/TRAVELCCM_Types.hpp>

Classes

- class TRAVELCCM::FacTRAVELCCMServiceContext

Namespaces

- namespace TRAVELCCM

25.48   FacTRAVELCCMServiceContext.hpp

```
00001 #ifndef __TRAVELCCM_FAC_FACTRAVELCCMSERVICECONTEXT_HPP
00002 #define __TRAVELCCM_FAC_FACTRAVELCCMSERVICECONTEXT_HPP
00003
00004 // //////////////////////////////////////////////////////////////////////
00005 // Import section
00006 // //////////////////////////////////////////////////////////////////////
00007 // StdAir
00008 #include <stdair/service/FacServiceAbstract.hpp>
00009 // TravelCCM
00010 #include <travelccm/TRAVELCCM_Types.hpp>
00011
00012 namespace TRAVELCCM {
00013
00015   class TRAVELCCM_ServiceContext;
00016
00018   class FacTRAVELCCMServiceContext : public stdair::FacServiceAbstract {
00019   public:
00020
00024     static FacTRAVELCCMServiceContext& instance();
00025
00030     ~FacTRAVELCCMServiceContext();
00031
00035     TRAVELCCM_ServiceContext& create ();
00036
00037
00038   protected:
00042     FacTRAVELCCMServiceContext () {}
00043
00044   private:
00046     static FacTRAVELCCMServiceContext* _instance;
00047   };
00048
00049 }
00050 #endif // __TRAVELCCM_FAC_FACTRAVELCCMSERVICECONTEXT_HPP
```

25.49   travelccm/service/TRAVELCCM_Service.cpp File Reference

```
#include <cassert>

#include <boost/make_shared.hpp>

#include <stdair/basic/BasChronometer.hpp>

#include <stdair/basic/BasFileMgr.hpp>

#include <stdair/bom/BomManager.hpp>

#include <stdair/bom/BookingRequestStruct.hpp>

#include <stdair/factory/FacBomManager.hpp>

#include <stdair/service/Logger.hpp>

#include <stdair/STDAIR_Service.hpp>

#include <travelccm/factory/FacTRAVELCCMServiceContext.hpp>

#include <travelccm/command/ChoiceManager.hpp>
```

```
#include <travelccm/service/TRAVELCCM_ServiceContext.hpp>
#include <travelccm/TRAVELCCM_Service.hpp>
```

Namespaces

- namespace TRAVELCCM

25.50 TRAVELCCM_Service.cpp

```
00001 // //////////////////////////////////////////////////////////////////////
00002 // Import section
00003 // //////////////////////////////////////////////////////////////////////
00004 // STL
00005 #include <cassert>
00006 // Boost
00007 #include <boost/make_shared.hpp>
00008 // StdAir
00009 #include <stdair/basic/BasChronometer.hpp>
00010 #include <stdair/basic/BasFileMgr.hpp>
00011 #include <stdair/bom/BomManager.hpp>
00012 #include <stdair/bom/BookingRequestStruct.hpp>
00013 #include <stdair/factory/FacBomManager.hpp>
00014 #include <stdair/service/Logger.hpp>
00015 #include <stdair/STDAIR_Service.hpp>
00016 // TravelCCM
00017 #include <travelccm/factory/FacTRAVELCCMServiceContext.hpp>
00018 #include <travelccm/command/ChoiceManager.hpp>
00019 #include <travelccm/service/TRAVELCCM_ServiceContext.hpp>
00020 #include <travelccm/TRAVELCCM_Service.hpp>
00021
00022 namespace TRAVELCCM {
00023
00024   // //////////////////////////////////////////////////////////////////////
00025   TRAVELCCM_Service::TRAVELCCM_Service() : _travelccmServiceContext (NULL) {
00026     assert (false);
00027   }
00028
00029   // //////////////////////////////////////////////////////////////////////
00030   TRAVELCCM_Service::TRAVELCCM_Service (const TRAVELCCM_Service& iService)
00031   : _travelccmServiceContext (NULL) {
00032     assert (false);
00033   }
00034
00035   // //////////////////////////////////////////////////////////////////////
00036   TRAVELCCM_Service::TRAVELCCM_Service (const stdair::BasLogParams& iLogParams,
00037                                         const stdair::BasDBParams& iDBParams)
00038     : _travelccmServiceContext (NULL) {
00039
00040     // Initialise the STDAIR service handler
00041     stdair::STDAIR_ServicePtr_T lSTDAIR_Service_ptr =
00042       initStdAirService (iLogParams, iDBParams);
00043
00044     // Initialise the service context
00045     initServiceContext();
00046
00047     // Add the StdAir service context to the AIRINV service context
00048     // \note AIRINV owns the STDAIR service resources here.
00049     const bool ownStdairService = true;
```

```
00050     addStdAirService (lSTDAIR_Service_ptr, ownStdairService);
00051
00052     // Initialise the (remaining of the) context
00053     initTravelCCMService();
00054   }
00055
00056   // //////////////////////////////////////////////////////////////////////
00057   TRAVELCCM_Service::TRAVELCCM_Service (const stdair::BasLogParams& iLogParams)
00058     : _travelccmServiceContext (NULL) {
00059
00060     // Initialise the STDAIR service handler
00061     stdair::STDAIR_ServicePtr_T lSTDAIR_Service_ptr =
00062       initStdAirService (iLogParams);
00063
00064     // Initialise the service context
00065     initServiceContext();
00066
00067     // Add the StdAir service context to the AIRINV service context
00068     // \note AIRINV owns the STDAIR service resources here.
00069     const bool ownStdairService = true;
00070     addStdAirService (lSTDAIR_Service_ptr, ownStdairService);
00071
00072     // Initialise the (remaining of the) context
00073     initTravelCCMService();
00074   }
00075
00076   // //////////////////////////////////////////////////////////////////////
00077   TRAVELCCM_Service::
00078   TRAVELCCM_Service (stdair::STDAIR_ServicePtr_T ioSTDAIR_Service_ptr)
00079     : _travelccmServiceContext (NULL) {
00080
00081     // Initialise the service context
00082     initServiceContext();
00083
00084     // Store the STDAIR service object within the (AIRINV) service context
00085     // \note AirInv does not own the STDAIR service resources here.
00086     const bool doesNotOwnStdairService = false;
00087     addStdAirService (ioSTDAIR_Service_ptr, doesNotOwnStdairService);
00088
00089     // Initialise the (remaining of the) context
00090     initTravelCCMService();
00091   }
00092
00093   // //////////////////////////////////////////////////////////////////////
00094   TRAVELCCM_Service::~TRAVELCCM_Service() {
00095     // Delete/Clean all the objects from memory
00096     finalise();
00097   }
00098
00099   // //////////////////////////////////////////////////////////////////////
00100   void TRAVELCCM_Service::finalise() {
00101     assert (_travelccmServiceContext != NULL);
00102     // Reset the (Boost.)Smart pointer pointing on the STDAIR_Service object.
00103     _travelccmServiceContext->reset();
00104   }
00105
00106   // //////////////////////////////////////////////////////////////////////
00107   void TRAVELCCM_Service::initServiceContext() {
00108     // Initialise the context
00109     TRAVELCCM_ServiceContext& lTRAVELCCM_ServiceContext =
00110       FacTRAVELCCMServiceContext::instance().create();
00111     _travelccmServiceContext = &lTRAVELCCM_ServiceContext;
```

```
00112    }
00113
00114    // //////////////////////////////////////////////////////////////////
00115    void TRAVELCCM_Service::
00116    addStdAirService (stdair::STDAIR_ServicePtr_T ioSTDAIR_Service_ptr,
00117                      const bool iOwnStdairService) {
00118      // Retrieve the Travelccm service context
00119      assert (_travelccmServiceContext != NULL);
00120      TRAVELCCM_ServiceContext& lTRAVELCCM_ServiceContext =
00121        *_travelccmServiceContext;
00122
00123      // Store the STDAIR service object within the (TRAVELCCM) service context
00124      lTRAVELCCM_ServiceContext.setSTDAIR_Service (ioSTDAIR_Service_ptr,
00125                                                   iOwnStdairService);
00126    }
00127
00128    // //////////////////////////////////////////////////////////////////
00129    stdair::STDAIR_ServicePtr_T TRAVELCCM_Service::
00130    initStdAirService (const stdair::BasLogParams& iLogParams,
00131                       const stdair::BasDBParams& iDBParams) {
00132
00139      stdair::STDAIR_ServicePtr_T oSTDAIR_Service_ptr =
00140        boost::make_shared<stdair::STDAIR_Service> (iLogParams, iDBParams);
00141      assert (oSTDAIR_Service_ptr != NULL);
00142
00143      return oSTDAIR_Service_ptr;
00144    }
00145
00146    // //////////////////////////////////////////////////////////////////
00147    stdair::STDAIR_ServicePtr_T TRAVELCCM_Service::
00148    initStdAirService (const stdair::BasLogParams& iLogParams) {
00149
00156      stdair::STDAIR_ServicePtr_T oSTDAIR_Service_ptr =
00157        boost::make_shared<stdair::STDAIR_Service> (iLogParams);
00158      assert (oSTDAIR_Service_ptr != NULL);
00159
00160      return oSTDAIR_Service_ptr;
00161    }
00162
00163    // //////////////////////////////////////////////////////////////////
00164    void TRAVELCCM_Service::initTravelCCMService() {
00165      // Do nothing at this stage. A sample BOM tree may be built by
00166      // calling the buildSampleBom() method
00167    }
00168
00169    // //////////////////////////////////////////////////////////////////
00170    void TRAVELCCM_Service::buildSampleBom() {
00171
00172      // Retrieve the TravelCCM service context
00173      if (_travelccmServiceContext == NULL) {
00174        throw stdair::NonInitialisedServiceException ("The TravelCCM service has "
00175                                                      "not been initialised");
00176      }
00177      assert (_travelccmServiceContext != NULL);
00178
00179      // Retrieve the TravelCCM service context and whether it owns the Stdair
00180      // service
00181      TRAVELCCM_ServiceContext& lTRAVELCCM_ServiceContext =
00182        *_travelccmServiceContext;
00183      const bool doesOwnStdairService =
00184        lTRAVELCCM_ServiceContext.getOwnStdairServiceFlag();
00185
```

```
00186      // Retrieve the StdAir service object from the (TravelCCM) service context
00187      stdair::STDAIR_Service& lSTDAIR_Service =
00188        lTRAVELCCM_ServiceContext.getSTDAIR_Service();
00189
00194      if (doesOwnStdairService == true) {
00195        //
00196        lSTDAIR_Service.buildSampleBom();
00197      }
00198
00213    }
00214
00215    // //////////////////////////////////////////////////////////////////////
00216    void TRAVELCCM_Service::
00217    buildSampleTravelSolutions (stdair::TravelSolutionList_T& ioTSList) {
00218
00219      // Retrieve the TRAVELCCM service context
00220      if (_travelccmServiceContext == NULL) {
00221        throw stdair::NonInitialisedServiceException ("The Travelccm service has "
00222                                                      "not been initialised");
00223      }
00224      assert (_travelccmServiceContext != NULL);
00225
00226      TRAVELCCM_ServiceContext& lTRAVELCCM_ServiceContext =
00227        *_travelccmServiceContext;
00228
00229      // Retrieve the STDAIR service object from the (TRAVELCCM) service context
00230      stdair::STDAIR_Service& lSTDAIR_Service =
00231        lTRAVELCCM_ServiceContext.getSTDAIR_Service();
00232
00233      // Delegate the BOM building to the dedicated service
00234      lSTDAIR_Service.buildSampleTravelSolutions (ioTSList);
00235    }
00236
00237    // //////////////////////////////////////////////////////////////////////
00238    stdair::BookingRequestStruct TRAVELCCM_Service::
00239    buildSampleBookingRequest (const bool isForCRS) {
00240
00241      // Retrieve the TRAVELCCM service context
00242      if (_travelccmServiceContext == NULL) {
00243        throw stdair::NonInitialisedServiceException ("The Travelccm service has "
00244                                                      "not been initialised");
00245      }
00246      assert (_travelccmServiceContext != NULL);
00247
00248      TRAVELCCM_ServiceContext& lTRAVELCCM_ServiceContext =
00249        *_travelccmServiceContext;
00250
00251      // Retrieve the STDAIR service object from the (TRAVELCCM) service context
00252      stdair::STDAIR_Service& lSTDAIR_Service =
00253        lTRAVELCCM_ServiceContext.getSTDAIR_Service();
00254
00255      // Delegate the BOM building to the dedicated service
00256      return lSTDAIR_Service.buildSampleBookingRequest (isForCRS);
00257    }
00258
00259    // //////////////////////////////////////////////////////////////////////
00260    std::string TRAVELCCM_Service::csvDisplay() const {
00261
00262      // Retrieve the TRAVELCCM service context
00263      if (_travelccmServiceContext == NULL) {
00264        throw stdair::NonInitialisedServiceException ("The TravelccmMaster service"
```

```
00265                                                       " has not been initialised");
00266     }
00267     assert (_travelccmServiceContext != NULL);
00268
00269     TRAVELCCM_ServiceContext& lTRAVELCCM_ServiceContext =
00270       *_travelccmServiceContext;
00271
00272     // Retrieve the STDAIR service object from the (TRAVELCCM) service context
00273     stdair::STDAIR_Service& lSTDAIR_Service =
00274       lTRAVELCCM_ServiceContext.getSTDAIR_Service();
00275
00276     // Delegate the BOM building to the dedicated service
00277     return lSTDAIR_Service.csvDisplay();
00278   }
00279
00280   // //////////////////////////////////////////////////////////////////////
00281   std::string TRAVELCCM_Service::
00282   csvDisplay (const stdair::TravelSolutionList_T& iTravelSolutionList) const {
00283     // Retrieve the TRAVELCCM service context
00284     if (_travelccmServiceContext == NULL) {
00285       throw stdair::NonInitialisedServiceException ("The TravelccmMaster service"
00286                                                       " has not been initialised");
00287     }
00288     assert (_travelccmServiceContext != NULL);
00289
00290     TRAVELCCM_ServiceContext& lTRAVELCCM_ServiceContext =
00291       *_travelccmServiceContext;
00292
00293     // Retrieve the STDAIR service object from the (TRAVELCCM) service context
00294     stdair::STDAIR_Service& lSTDAIR_Service =
00295       lTRAVELCCM_ServiceContext.getSTDAIR_Service();
00296
00297     // Delegate the BOM building to the dedicated service
00298     return lSTDAIR_Service.csvDisplay (iTravelSolutionList);
00299   }
00300
00301   // //////////////////////////////////////////////////////////////////////
00302   const stdair::TravelSolutionStruct* TRAVELCCM_Service::
00303   chooseTravelSolution (stdair::TravelSolutionList_T& ioTravelSolutionList,
00304                         const stdair::BookingRequestStruct& iBookingRequest) {
00305
00306     const stdair::TravelSolutionStruct* oTravelSolution_ptr =
00307       ChoiceManager::chooseTravelSolution (ioTravelSolutionList,
00308                                            iBookingRequest);
00309     return oTravelSolution_ptr;
00310   }
00311
00312 }
```

25.51 travelccm/service/TRAVELCCM_ServiceContext.cpp File Reference

#include <cassert>

#include <sstream>

#include <travelccm/service/TRAVELCCM_ServiceContext.hpp>

Namespaces

- namespace TRAVELCCM

25.52 TRAVELCCM_ServiceContext.cpp

```
00001  // //////////////////////////////////////////////////////////////////////
00002  // Import section
00003  // //////////////////////////////////////////////////////////////////////
00004  // STL
00005  #include <cassert>
00006  #include <sstream>
00007  // TravelCCM Basic
00008  #include <travelccm/service/TRAVELCCM_ServiceContext.hpp>
00009
00010  namespace TRAVELCCM {
00011
00012    // //////////////////////////////////////////////////////////////////////
00013    TRAVELCCM_ServiceContext::TRAVELCCM_ServiceContext()
00014      : _ownStdairService (false) {
00015    }
00016
00017    // //////////////////////////////////////////////////////////////////////
00018    TRAVELCCM_ServiceContext::
00019    TRAVELCCM_ServiceContext (const TRAVELCCM_ServiceContext&) {
00020      assert (false);
00021    }
00022
00023    // //////////////////////////////////////////////////////////////////////
00024    TRAVELCCM_ServiceContext::~TRAVELCCM_ServiceContext() {
00025    }
00026
00027    // //////////////////////////////////////////////////////////////////////
00028    const std::string TRAVELCCM_ServiceContext::shortDisplay() const {
00029      std::ostringstream oStr;
00030      oStr << "TRAVELCCM_ServiceContext -- Owns StdAir service: "
00031           << _ownStdairService;
00032      return oStr.str();
00033    }
00034
00035    // //////////////////////////////////////////////////////////////////////
00036    const std::string TRAVELCCM_ServiceContext::display() const {
00037      std::ostringstream oStr;
00038      oStr << shortDisplay();
00039      return oStr.str();
00040    }
00041
00042    // //////////////////////////////////////////////////////////////////////
00043    const std::string TRAVELCCM_ServiceContext::describe() const {
00044      return shortDisplay();
00045    }
00046
00047    // //////////////////////////////////////////////////////////////////////
00048    void TRAVELCCM_ServiceContext::reset() {
00049      if (_ownStdairService == true) {
00050        _stdairService.reset();
00051      }
00052    }
00053
00054  }
```

25.53 travelccm/service/TRAVELCCM_ServiceContext.hpp File Reference

```
#include <string>

#include <boost/shared_ptr.hpp>

#include <stdair/stdair_service_types.hpp>

#include <stdair/service/ServiceAbstract.hpp>

#include <travelccm/TRAVELCCM_Types.hpp>
```

Classes

- class TRAVELCCM::TRAVELCCM_ServiceContext

  Inner class holding the context for the TravelCCM Service object.

Namespaces

- namespace TRAVELCCM

25.54 TRAVELCCM_ServiceContext.hpp

```
00001 #ifndef __TRAVELCCM_SVC_TRAVELCCM_SERVICE_CONTEXT_HPP
00002 #define __TRAVELCCM_SVC_TRAVELCCM_SERVICE_CONTEXT_HPP
00003
00004 // //////////////////////////////////////////////////////////////////////
00005 // Import section
00006 // //////////////////////////////////////////////////////////////////////
00007 // STL
00008 #include <string>
00009 // Boost
00010 #include <boost/shared_ptr.hpp>
00011 // StdAir
00012 #include <stdair/stdair_service_types.hpp>
00013 #include <stdair/service/ServiceAbstract.hpp>
00014 // TravelCCM
00015 #include <travelccm/TRAVELCCM_Types.hpp>
00016
00017 namespace TRAVELCCM {
00018
00023   class TRAVELCCM_ServiceContext : public stdair::ServiceAbstract {
00029     friend class TRAVELCCM_Service;
00030     friend class FacTRAVELCCMServiceContext;
00031
00032   private:
00033     // //////////////// Getters ////////////////
00037     stdair::STDAIR_ServicePtr_T getSTDAIR_ServicePtr() const {
00038       return _stdairService;
00039     }
00040
00044     stdair::STDAIR_Service& getSTDAIR_Service() const {
00045       assert (_stdairService != NULL);
00046       return *_stdairService;
00047     }
00048
```

```
00052     const bool getOwnStdairServiceFlag() const {
00053       return _ownStdairService;
00054     }
00055
00056
00057     // ///////////////// Setters //////////////////
00061     void setSTDAIR_Service (stdair::STDAIR_ServicePtr_T ioSTDAIR_ServicePtr,
00062                             const bool iOwnStdairService) {
00063       _stdairService = ioSTDAIR_ServicePtr;
00064       _ownStdairService = iOwnStdairService;
00065     }
00066
00067
00068   private:
00069     // /////////////////// Display Methods /////////////////////
00073     const std::string shortDisplay() const;
00074
00078     const std::string display() const;
00079
00083     const std::string describe() const;
00084
00085
00086   private:
00088
00091     TRAVELCCM_ServiceContext();
00095     TRAVELCCM_ServiceContext (const TRAVELCCM_ServiceContext&);
00096
00100     ~TRAVELCCM_ServiceContext();
00101
00105     void reset();
00106
00107
00108   private:
00109     // /////////////// Children ///////////////
00113     stdair::STDAIR_ServicePtr_T _stdairService;
00114
00118     bool _ownStdairService;
00119   };
00120
00121 }
00122 #endif // __TRAVELCCM_SVC_TRAVELCCM_SERVICE_CONTEXT_HPP
```

25.55 travelccm/TRAVELCCM_Service.hpp File Reference

```
#include <stdair/stdair_basic_types.hpp>

#include <stdair/stdair_service_types.hpp>

#include <stdair/bom/TravelSolutionTypes.hpp>

#include <travelccm/TRAVELCCM_Types.hpp>
```

Classes

- class TRAVELCCM::TRAVELCCM_Service

25.56   TRAVELCCM_Service.hpp

```
00001 #ifndef __TRAVELCCM_SVC_TRAVELCCM_SERVICE_HPP
00002 #define __TRAVELCCM_SVC_TRAVELCCM_SERVICE_HPP
00003
00004 // //////////////////////////////////////////////////////////////////////
00005 // Import section
00006 // //////////////////////////////////////////////////////////////////////
00007 // StdAir
00008 #include <stdair/stdair_basic_types.hpp>
00009 #include <stdair/stdair_service_types.hpp>
00010 #include <stdair/bom/TravelSolutionTypes.hpp>
00011 // TravelCCM
00012 #include <travelccm/TRAVELCCM_Types.hpp>
00013
00015 namespace stdair {
00016   class STDAIR_Service;
00017   struct BasLogParams;
00018   struct BasDBParams;
00019   struct BookingRequestStruct;
00020 }
00021
00022 namespace TRAVELCCM {
00023
00025   class TRAVELCCM_ServiceContext;
00026
00027
00031   class TRAVELCCM_Service {
00032   public:
00033     // ////////// Constructors and destructors //////////
00049     TRAVELCCM_Service (const stdair::BasLogParams&, const stdair::BasDBParams&);
00050
00062     TRAVELCCM_Service (const stdair::BasLogParams&);
00063
00079     TRAVELCCM_Service (stdair::STDAIR_ServicePtr_T);
00080
00084     ~TRAVELCCM_Service();
00085
00086
00087   public:
00088     // //////////// Business methods ////////////////
00092     void buildSampleBom();
00093
00112     void buildSampleTravelSolutions (stdair::TravelSolutionList_T&);
00113
00142     stdair::BookingRequestStruct
00143     buildSampleBookingRequest (const bool isForCRS = false);
00144
00159     const stdair::TravelSolutionStruct*
00160     chooseTravelSolution (stdair::TravelSolutionList_T&,
00161                           const stdair::BookingRequestStruct&);
00162
00163
```

```
00164   public:
00165     // ///////////////// Display support methods //////////////////
00173     std::string csvDisplay() const;
00174
00182     std::string csvDisplay (const stdair::TravelSolutionList_T&) const;
00183
00184
00185   private:
00186     // /////// Construction and Destruction helper methods ///////
00190     TRAVELCCM_Service();
00194     TRAVELCCM_Service (const TRAVELCCM_Service&);
00195
00205     stdair::STDAIR_ServicePtr_T initStdAirService (const stdair::BasLogParams&,
00206                                                    const stdair::BasDBParams&);
00207
00216     stdair::STDAIR_ServicePtr_T initStdAirService (const stdair::BasLogParams&);
00217
00226     void addStdAirService (stdair::STDAIR_ServicePtr_T ioSTDAIR_ServicePtr,
00227                            const bool iOwnStdairService);
00228
00233     void initServiceContext();
00234
00241     void initTravelCCMService();
00242
00246     void finalise();
00247
00248   private:
00249     // ///////// Service Context /////////
00253     TRAVELCCM_ServiceContext* _travelccmServiceContext;
00254   };
00255 }
00256 #endif // __TRAVELCCM_SVC_TRAVELCCM_SERVICE_HPP
```

25.57 travelccm/TRAVELCCM_Types.hpp File Reference

```
#include <string>

#include <boost/shared_ptr.hpp>

#include <stdair/stdair_exceptions.hpp>
```

Classes

- class TRAVELCCM::CustomerChoiceException

Namespaces

- namespace TRAVELCCM

Typedefs

- typedef boost::shared_ptr< TRAVELCCM_Service > TRAVELCCM::TRAVELCCM_-
  ServicePtr_T

25.58 TRAVELCCM␣Types.hpp

```
00001 #ifndef __TRAVELCCM_TRAVELCCM_TYPES_HPP
00002 #define __TRAVELCCM_TRAVELCCM_TYPES_HPP
00003
00004 // //////////////////////////////////////////////////////////////////////
00005 // Import section
00006 // //////////////////////////////////////////////////////////////////////
00007 // STL
00008 #include <string>
00009 // Boost
00010 #include <boost/shared_ptr.hpp>
00011 // StdAir
00012 #include <stdair/stdair_exceptions.hpp>
00013
00014 namespace TRAVELCCM {
00015
00017   class TRAVELCCM_Service;
00018
00019
00020   // ///////// Exceptions ///////////
00024   class CustomerChoiceException : public stdair::RootException {
00025   public:
00029     CustomerChoiceException (const std::string& iWhat)
00030       : stdair::RootException (iWhat) {}
00031   };
00032
00033
00034   // //////// Type definitions /////////
00038   typedef boost::shared_ptr<TRAVELCCM_Service> TRAVELCCM_ServicePtr_T;
00039
00040 }
00041 #endif // __TRAVELCCM_TRAVELCCM_TYPES_HPP
```

Index