



MAK ONE



Copyright © 2025 MAK Technologies
All rights Reserved. Printed in the United States.

Under copyright laws, no part of this document may be copied or reproduced in any form without prior written consent of MAK Technologies.

VR-Exchange™, VR-TheWorld™, VR-Vantage™, DI-Guy™, and DI-Guy Scenario™ are trademarks of MAK Technologies. MAK Technologies®, VR-Forces®, RTIsPy®, B-HAVE®, and VR-Link® are registered trademarks of MAK Technologies.

GL Studio® is a registered trademark of The DiSTI® Corporation.

Portions of this software utilize SpeedTree® technology (©2008 Interactive Data Visualization, Inc.). SpeedTree is a registered trademark of Interactive Data Visualization, Inc. All rights reserved.

SilverLining™ is a trademark of Sundog Software.

Terrain Profiles are based in part on the work of the Qwt project (<http://qwt.sourceforge.net>).

3ds Max® is a registered trademark of Autodesk, Inc.

All other trademarks are owned by their respective companies.

For third-party license information, see "Third-Party Licenses" on page 9 in the Preface.

MAK Technologies
10 Fawcett Street, Suite 204
Cambridge, MA 02138 USA

Voice: +1 617.876.8085

Fax: +1 617.876.9208

info@mak.com

www.mak.com

Revision RFV-1.0-02-250408



Contents

Preface

MAK Products	6
How to Contact Us	8
Telephone	8
E-mail	8
Internet	8
Post	8
Document Conventions	8
Mouse Button Naming Conventions	9
Third-Party Licenses	9
Third-Party Libraries Supported	9
Library Licensing	10
Boost License	12
libXML and libICONV	13

1. Introduction to RFView® for MAK ONE

1.1. Overview of RFView for MAK ONE	16
1.1.1 Product Interactions and Data Flow	16
1.2. Installing RFView® for MAK ONE	17
1.2.1 Installation and Data Packages	17
1.2.2 Installing RFView for MAK ONE on Windows	17
1.2.3 Installing RFView for MAK ONE PDB Files for Debugging Purposes	18
1.2.4 The makData Package	20
1.3. The RFView for MAK ONE Directory Structure	22
1.4. Installing and Setting Up the MAK License Manager	23
1.4.1 Specifying the License Server	23
1.5. Installing an RTI	25
1.5.1 Installing the MAK RTI	26

1.6. Localizing the Graphical User Interface	26
1.6.1 Translating Other Interface Files	29
1.6.2 Applying the Language Files	29
 2. Using RFView® for MAK ONE Server	
2.1. Starting RFView for MAK ONE Server	32
2.2. Connecting to a Simulation	32
 3. Using the RFView® for MAK ONE Simple Client	
3.1. Starting RFView® Simple Client	34
3.2. Requesting a SAR Image	34
3.2.1 Specifying the Location of the Sensor	35
3.2.2 Specifying the Area of Interest	36
3.2.3 Sending the Request	36
3.3. Configuring Image Properties	36
 Index	39



Preface

This manual is for people who will install RFView® for MAK ONE and use it to view simulations. The manual assumes that you are familiar with performing basic administrative tasks in your operating system. Use of the VR-Vantage interface and material classification of models and terrain are outside the scope of this document. To learn about either of those, see the VR-Vantage help. You can also refer to the *VR-Vantage User Guide* for information about the former and the *Adding Content to MAK ONE Applications Reference Manual* for the latter.

For the latest product information, see the release-specific documentation for your version of RFView for MAK ONE.

MAK Products

The MAK Technologies product line includes:

- **VR-Link® Network Toolkit.** VR-Link is an object-oriented library of C++ functions and definitions that implement the High Level Architecture (HLA) and the Distributed Interactive Simulation (DIS) protocol. VR-Link has built-in support for the NETN FOM and RPR FOM and allows you to map to other FOMs. This library minimizes the time and effort required to build and maintain new HLA or DIS-compliant applications, and to integrate such compliance into existing applications.

VR-Link includes a set of sample debugging applications and their source code. The source code serves as an example of how to use the VR-Link Toolkit to write applications. The executables provide valuable debugging services such as generating a predictable stream of HLA or DIS messages, and displaying the contents of messages transmitted on the network.
- **MAK RTI.** An RTI (Run-Time Infrastructure) is required to run applications using the High Level Architecture (HLA). The MAK RTI is optimized for high performance. It has an API, RTIsby®, that allows you to extend the RTI using plug-in modules. It also has a graphical user interface (the RTI Assistant) that helps users with configuration tasks and managing federates and federations.
- **VR-Forces®.** VR-Forces is a computer generated forces application and toolkit. It provides an application with a GUI, that gives you 2D and 3D views of a simulated environment.

You can create and view entities, aggregate them into hierarchical units, assign tasks, set state parameters, and create plans that have tasks, set statements, and conditional statements. You can simulate using entity-level modeling, which focuses on the actions of individual people and vehicles, and aggregate-level modeling, which focuses on the interaction of large hierarchical units.

VR-Forces also functions as a plan view display for viewing remote simulation objects taking part in an exercise. Using the toolkit, you can extend the VR-Forces application or create your own application for use with another user interface.

- **VR-Vantage™.** VR-Vantage is a line of products designed to meet your simulation visualization needs. It includes VR-Vantage and the VR-Vantage Toolkit.
 - VR-Vantage displays a realistic, 3D view of your virtual world, a 2D plan view, and an exaggerated reality (XR) view. Together these views provide both situational awareness and the big picture of the simulated world. You can move your viewpoint to any location in the 3D world and can attach it to simulation objects so that it moves as they do.
 - **SensorFX.** SensorFX is an enhanced version of VR-Vantage that uses physics-based sensors to view terrain and simulation object models that have been materially classified. It is built in partnership with JRM Technologies.
 - The VR-Vantage Toolkit is a 3D visual application development toolkit. Use it to customize or extend MAK's VR-Vantage applications, or to integrate VR-

Vantage capabilities into your custom applications. VR-Vantage is built on top of OpenSceneGraph (OSG). The toolkit includes the OSG version used to build VR-Vantage.

- **MAK Data Logger.** The Data Logger, also called the Logger, can record HLA and DIS exercises and play them back for after-action review. You can play a recorded file at speeds above or below normal and can quickly jump to areas of interest. The Logger has a GUI and a text interface. The Logger API allows you to extend the Logger using plug-in modules or embed the Logger into your own application. The Logger editing features let you merge, trim, and offset Logger recordings.
- **VR-Exchange™.** VR-Exchange allows simulations that use incompatible communications protocols to interoperate. For example, within the HLA world, using VR-Exchange, federations using the HLA RPR FOM 1.0 can interoperate with simulations using RPR FOM 2.0, or federations using different RTIs can interoperate. VR-Exchange supports HLA, TENA, and DIS translation.
- **VR-TheWorld™.** VR-TheWorld is a simple, yet powerful, web-based streaming terrain server, developed in conjunction with Pelican Mapping. Delivered with a global base map, you can also easily populate it with your own custom source data through a web-based interface. The server can be deployed on private, classified networks to provide streaming terrain data to a variety of simulation and visualization applications behind your firewall.
- **DI-Guy™.** DI-Guy is an SDK that allows you to embed the DI-Guy library in your real-time application and populate your world with lifelike human characters. DI-Guy provides high-fidelity human characters that move realistically, respond to simple high-level commands, and travel throughout the environment as directed by the hosting visual application. It comes with thousands of ready-to-use characters, appearances, and motions. DI-Guy character simulation is an integral part of all MAK's visual applications, including VR-Engage, VR-Forces, and VR-Vantage.
- **RFView® for MAK ONE.** Synthetic Aperture Radar (SAR) imaging is an indispensable tool for today's modern militaries. Developed in partnership with ISL, RFView for MAK ONE acts as a SAR server and delivers simulated SAR images that are both physically accurate and fully correlated with out-the-window views and EO/IR images of MAK ONE synthetic environments. It includes a toolkit that enables you to build a client application to request SAR images from the server.

RFView® is a registered trademark of ISL, Inc.

- **VR-Engage.** VR-Engage is a multi-role virtual simulator that lets users play the role of a first person human character, a ground vehicle driver, gunner or commander, a sensor operator, or the pilot of a fixed wing aircraft or helicopter. It incorporates the VR-Force simulation engine and the VR-Vantage graphics rendering capabilities.
- **WebLVC Server.** WebLVC Server implements the server side of the WebLVC protocol so that web-based simulation federates can participate in a distributed simulation. It is part of the WebLVC Suite, which includes the server and several sample applications that work with VR-Forces and VR-Vantage.
- **MAK Legion.** MAK Legion is a collection of applications and an SDK designed to support simulations with millions of simulation objects.

How to Contact Us

For technical support, information about upgrades, and information about other MAK products, you can contact us in the following ways:

Telephone

Call or fax us at:	Voice:	617-876-8085
	Fax:	617-876-9208

E-mail

Sales and upgrade information:	info@mak.com
Technical support:	support@mak.com

Internet

MAK website home page:	www.mak.com
License key requests:	https://www.mak.com/mak-one/support/portal
Technical support:	https://www.mak.com/mak-one/support/portal
Product version and platform information:	https://www.mak.com/mak-one-specs
For the free, unlicensed MAK RTI:	https://www.mak.com/mak-one/support/help/bonus-material

Post

Send postal correspondence to:	MAK Technologies 10 Fawcett Street, Suite 204 Cambridge, MA 02138 USA
--------------------------------	---

When requesting support, tell us the product you are using, the version, and the platform on which you are running. If you are using VR-Vantage, VR-Forces, VR-Engage, or DI-Guy, please also include your data compatibility version and date.

Document Conventions

This manual uses these typographic conventions:

Sans Serif Bold	Indicates a key on the keyboard or menu options. Mentions of letter keys do not require you to press the Shift key. If Shift is necessary, that will be included, for example, Shift+Q .
Monospaced	Also indicates parameters.
<i>Monospaced</i>	Indicates values you enter such as commands or arguments.
<i>Italic</i>	Indicates command variables that you replace with appropriate values.
{ }	Indicates required arguments.
[]	Indicates optional arguments.
	Separates options in a command where only one option may be chosen at a time.

()	In command syntax, indicates equivalent alternatives for a command-line option, for example, (-h --help).
/	Indicates a directory. Since MAK products run on both Linux and Windows PC platforms, we use the / (slash) for generic discussions of pathnames.
<i>Italic</i>	Indicates a file name, pathname, or a class name.
►	Indicates a procedure.
Menu > Option	Indicates a menu choice. For example, an instruction to select the Save option from the File menu appears as: Choose File > Save .
	Indicates supplemental or clarifying information.
	Indicates additional information that you must observe to ensure the success of a procedure or other task.

Directory names are preceded with dot and slash characters that show their position with respect to the RFView for MAK ONE home directory. For example, the directory appears in the text as *./doc*.

Mouse Button Naming Conventions

An instruction to click the mouse button refers to clicking the primary mouse button, usually the left button for right-handed mice and the right button for left-handed mice. The context-sensitive menu, also called a popup menu or right-click menu, refers to the menu displayed when you click the secondary mouse button, usually the right button on right-handed mice and the left button on left-handed mice.

Third-Party Licenses

MAK software products may use code from third parties. This section contains summary license information and, where required, specific license documentation required by these third parties.

Third-Party Libraries Supported

RFView for MAK ONE 1.0 uses the indicated versions of these libraries:

- VR-Link 5.9.1
- Qt 5.15.2 (VC 15 and Red Hat 8)
- OpenSceneGraph 3.6.4
- Boost 1.84.0
- FreeGlut 2.4.0 (Windows), 2.6.0 (Linux)
- GL Studio 8.2
- DI-Guy 13.7.1 (development version)
- Triton 4.43
- SilverLining 6.20
- SpeedTree 6.3.2

- CDB API 2.2 (Windows), 2.1 (Linux)
- osgEarth 3.2
- CUDA 12.5 (Windows), 12.4.1 (Linux)

Library Licensing

These libraries are covered by the GNU Lesser General Public License (LGPL) license:

- CIGI
- DevIL
- GEOS
- GStreamer
- LibIconV
- LibWebSockets
- OPCode
- OpenAL
- OSG
- OsgEarth
- Pthread
- Qt
- Qwt

These libraries are covered by the ZLib license:

- Bullet Physics
- LibPNG
- LibSDL
- Minizip
- Zlib

These libraries are covered by the MIT license:

- Expat
- FreeGLUT
- GDAL
- GeoTIFF
- GLM
- HarfBuzz
- LibCURL
- LibSquish
- LibXML
- LUA
- LuaBind
- Proj
- SMHasher
- TCLAP
- Qt Visual Graph Editor
- ZeroBrane Studio

These libraries are covered by the BSD license:

- FreeType
- GLEW
- JPEG
- LibTiff
- Protobuf
- PCRE
- Ogg Vorbis
- WebP

These libraries are covered by the commercial licenses:

- FLEXlm
- Gameware
- GLStudio
- JRM Technologies SenSimRT and SigSimRT
- OpenFlight
- SpeedTree
- Sundog Triton
- Sundog SilverLining
- CDB API

These libraries are covered by custom licenses:

- NVIDIA CG Toolkit
- FreeImage (FreeImage Public License)
- Boost (Boost Software License)
- Poco (Boost Software License)

COLLADA DOM is covered by the SCEA Shared Source license.

Thread Building Blocks is covered by the GPL license:

SQLite is in the public domain.

SWIG is covered by the GPL license.

Boost License

VR-Link, and all MAK ONE products that use VR-Link, uses some code that is distributed under the Boost License. All header files that contain Boost code are properly attributed. The Boost website is: www.boost.org.

Boost Software License - Version 1.63 - August 17th, 2003

Permission is hereby granted, free of charge, to any person or organization obtaining a copy of the software and accompanying documentation covered by this license (the “Software”) to use, reproduce, display, distribute, execute, and transmit the Software, and to prepare derivative works of the Software, and to permit third parties to whom the Software is furnished to do so, all subject to the following:

The copyright notices in the Software and this entire statement, including the above license grant, this restriction and the following disclaimer, must be included in all copies of the Software, in whole or in part, and all derivative works of the Software, unless such copies or derivative works are solely in the form of machine-executable object code generated by a source language processor.

THE SOFTWARE IS PROVIDED “AS IS”, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT. IN NO EVENT SHALL THE COPYRIGHT HOLDERS OR ANYONE DISTRIBUTING THE SOFTWARE BE

LIABLE FOR ANY DAMAGES OR OTHER LIABILITY, WHETHER IN CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

libXML and libICONV

VR-Link, and all MAK ONE products that use VR-Link, links in libXML and libICONV. On some platforms the compiled libraries and header files are distributed with MAK products. MAK has made no modifications to these libraries. For more information about these libraries, see the following websites:

- The LGPL license is available at: <http://www.gnu.org/licenses/lgpl.html>.
- Information about IconV is at: <http://www.gnu.org/software/libiconv>.
- Information about LibXML is at: <http://xmlsoft.org>.



1. Introduction to RFView® for MAK ONE

This chapter provides a brief introduction to RFView for MAK ONE and explains how to install it. You must also install the MAK license manager. To use RFView for MAK ONE with HLA, you must install an RTI.

1.1. Overview of RFView for MAK ONE	16
1.1.1 Product Interactions and Data Flow	16
1.2. Installing RFView® for MAK ONE	17
1.2.1 Installation and Data Packages	17
1.2.2 Installing RFView for MAK ONE on Windows	17
1.2.3 Installing RFView for MAK ONE PDB Files for Debugging Purposes	18
1.2.4 The makData Package	20
1.3. The RFView for MAK ONE Directory Structure	22
1.4. Installing and Setting Up the MAK License Manager	23
1.4.1 Specifying the License Server	23
1.5. Installing an RTI	25
1.5.1 Installing the MAK RTI	26
1.6. Localizing the Graphical User Interface	26
1.6.1 Translating Other Interface Files	29
1.6.2 Applying the Language Files	29

1.1. Overview of RFView for MAK ONE

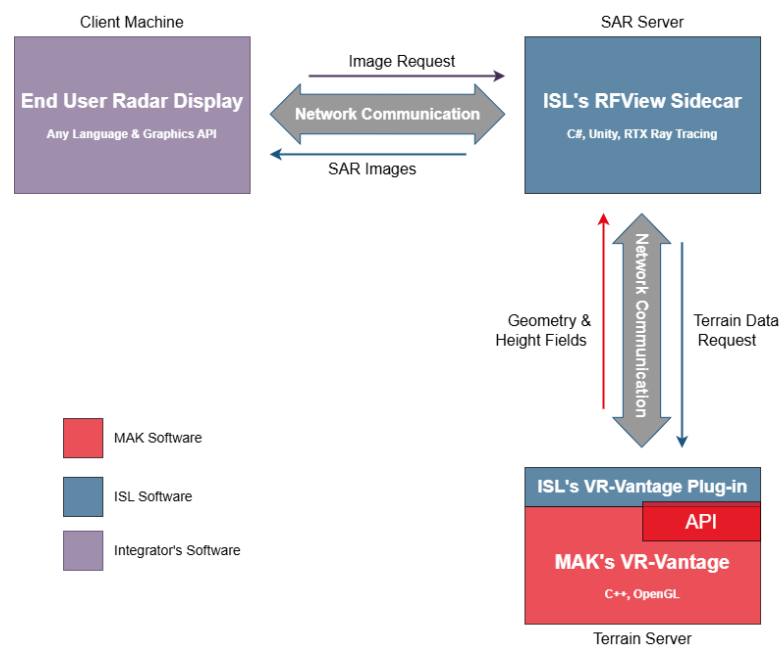
RFView for MAK ONE consists of a SAR server (RFView Sidecar) and a terrain server (VR-Vantage). It delivers high-fidelity simulated SAR images of MAK ONE synthetic environments with real-time GPU-based ray tracing. RFView for MAK ONE delivers credible, physics-based SAR effects—including layover, foreshortening, speckle, and multi-bounce effects not achievable with other approaches. RFView for MAK ONE 1.0 is based on VR-Vantage 3.1.1.

1.1.1 Product Interactions and Data Flow

RFView for MAK ONE is a collaboration between MAK and ISL. RFView for MAK ONE enables a client application to send a request through RFView Sidecar and receive a SAR image from the server of the scene shown in VR-Vantage.

- The client application is the end user radar display, for example, an instructor operator station, a radar training pod, or a flight simulator. RFView for MAK ONE includes the RFView Simple Client, which demonstrates the basic functionality of such an application.
- The communication between the SAR server (RFView Sidecar) and the terrain server (VR-Vantage):
 - Uses a socket interface.
 - Does not need to rebuild when one side changes.
 - Avoids language compatibility issues.
 - Can be different computers or a local loopback.
- The client machine and SAR server are typically different computers.

Figure 1. Data Flow



The flow of data is:

1. The user of the radar display application on the client machine configures the sensor location and then requests an image from the SAR server.
2. RFView Sidecar on the SAR server receives the requests and in turn requests the terrain data from VR-Vantage on the terrain server.
3. VR-Vantage returns the geometry and height field information to RFView Sidecar.
4. RFView Sidecar sends the SAR image to the radar display.

1.2. Installing RFView® for MAK ONE

1.2.1 Installation and Data Packages

MAK ONE installations are provided in one or more files (packages) that you can download from the MAK website.

There are four types of installation products for MAK ONE:

- Product installers – Each product has a single main installer for each operating system.
- Optional installers – Some products have installers for developer options.
- Required data packages – MAK ONE applications (VR-Vantage, VR-Forces, VR-Engage, SensorFX, and RadarFX) and the DI-Guy SDK have a collection of data packages that are required for those products' basic functions.
- Recommended data packages – Additional data packages (if available) are recommended to access the full capabilities of MAK ONE applications.

As of spring 2022, new releases of MAK ONE applications share the same data installation, therefore the same set of data packages is all you need for any or all applications (VR-Vantage, VR-Forces, VR-Engage, SensorFX, and RadarFX).

Dependencies

Licensing tools and dependent products have their own installers. For details, see .

For interoperability using HLA, see

Before you install RFView® for MAK ONE, please read the release notes to see if there are any special instructions for installation.

1.2.2 Installing RFView for MAK ONE on Windows

Windows versions of RFView for MAK ONE are provided as downloaded files.

- To install RFView for MAK ONE, run the installer. Follow the instructions in the installation wizard. You install the data used by RFView for MAK ONE separately, as described in "1.2.4 The makData Package" on page 20.)

Note the following:

- The installers are named to indicate the compiler used to build that version of RFView for MAK ONE.
- When you install large applications on Windows, there may be a delay of up to several minutes from the time you try to run the setup program to the time that an installation dialog box displays. This is due to how Windows scans setup programs before it executes them. If you experience this problem, turning off User Access Control can reduce or eliminate this delay.

Default Windows Installation Directory

The default root directory for installing MAK ONE products is *C:/MAK*.

You can use an environment variable, `MAK_INSTALL_DIR`, to specify a different location where all MAK ONE products and data should be installed by default. For example, if you define it as `MAK_INSTALL_DIR = D:\MyProject`, then all MAK ONE products automatically install to that location. The data packages and products would then install in that directory by default: *D:/MyProject/SharedData*, *D:/MyProject/vrlink5.9.1*, and so on.

Uninstalling from Windows

MAK ONE product installations include an uninstaller application (*unins000.exe*), which you can find in the installation directory. Run that program to uninstall the product.

When you run other installers for the product, such as the PDBs, there are additional uninstallers. If you run the uninstallers from the highest number down (for example, *unins002.exe* and then *unins001.exe*), running *unins000.exe* uninstalls everything else and removes the install folder.

When you uninstall VR-Vantage, VR-Forces, or VR-Engage, files such as scenarios or SMSs that reside in the product directory are also deleted. Before uninstalling, copy any of your custom files from the product directory to another folder.

The uninstaller does not remove the data package. That is in a separate directory (for example, *./SharedData/18/latest*).

i If you run only the product uninstaller, you can then simply delete the rest of the product's install folder. Note that if you do not run all of the uninstallers, running only the product uninstaller may not remove all of the shortcuts from the Windows Start menu.

1.2.3 Installing RFView for MAK ONE PDB Files for Debugging Purposes

When a MAK ONE application crashes, it can be difficult to determine the cause. The application generates a *.dmp* file that contains valuable information about the application's state at the time of the crash. MAK technical support can analyze this file to help diagnose the issue.

In addition, with the application's PDB (program database) files, the application generates a call stack log file (*.callstack.log*) of the application at the time of the crash. The call stack log provides information in plain text about the sequence of functions that were called leading up to the crash, which is also helpful in identifying the specific cause.

By providing the *.dmp* or *.callstack.log* files to technical support, you help support personnel to resolve the issues. They use specialized debugging tools to analyze the files and determine the cause of the crash, and that information aids in the development of a solution to prevent future crashes.

Due to security reasons or organization policies, some customers cannot send the *.dmp* file to technical support. Because the call stack log file is a plain text file, it is easier to inspect and analyze than the *.dmp* file. Users can easily attach it to an email and pass it out of secure areas with visual inspection, making it a convenient option for customers with restrictions on sharing binary files.

It is important to note that the call stack log file alone might not be sufficient for support personnel to fully diagnose the issue. They may require more information to gain a complete understanding of the issue. A detailed procedure for reproducing the crash often helps them solve the problem you encountered.



- Support for the generation of the call stack log file was introduced in MAK ONE 2024. It is supported only in MAK ONE products part of that release and later versions.
- The PDB files are part of the Linux installation. You do not need to install them separately.

Installing PDB Files on Windows

Before installing the PDB files on Windows, you must install RFView for MAK ONE. Next, make sure that the PDB files you are installing are from the same build as the application. When you download the installer from the [MAK ONE Product Downloads](#) page, the appropriate PDB installer is also available for download.

► To install the PDB files:

1. After installing RFView for MAK ONE, run the PDB installer for that product.
2. The installer looks for the matching product directory by searching the registry and then it automatically sets the install directory to the appropriate path. If that is not correct for your product, see the note below.
3. Complete the installation wizard. The necessary PDB files are installed in the product's *./bin64* directory.

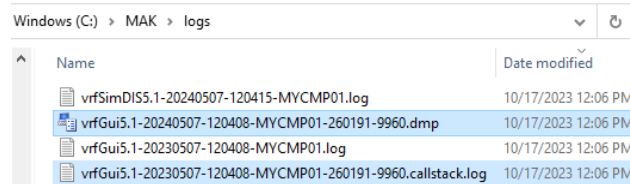


- The default installation directory is *C:\MAK*. See "Default Windows Installation Directory" on the previous page to learn about setting the default to other locations.
- Be sure that the folder in the Select Destination Location page of the installation wizard is correct. It should be the top-level folder for the product installation.
- If the PDB installer displays a warning that the SVN revisions do not match, make sure that you have the correct PDB installer for the version of that you installed.

Reporting an Issue Using the *.callstack.log* and *.dmp* Files

Send the *.callstack.log* and *.dmp* files (or only the *.callstack.log* if you are unable to provide the *.dmp* file) to MAK support through the Support Portal (<https://www.mak.com/mak-one/support/portal>) or support@mak.com. The files will have similar names and are located in the central directory for MAK ONE log files (Figure 2). Please also provide information on anything you experienced when the issue occurred.

Figure 2. Example of *.callstack.log* and *.dmp* files for VR-Forces



Specifying the Location for *.log* and *.dmp* Files

The default directory for MAK ONE log files is *C:/MAK/logs*. You can customize the location by specifying a different path in the MAK_LOG_DIR environment variable.

1.2.4 The makData Package

The MAK Data Packages are used by multiple MAK ONE applications. The applications need to be able to find the data, wherever you install it. RFView for MAK ONE looks for data in a default location or uses a collection of environment variables to find the data in the location where you installed it. You have several options for installing the data and telling RFView for MAK ONE where to look for it.

MAK Data Installation Directory

The shared data files are stored in a directory file hierarchy. This structure allows for multiple versions of data to be installed for use with different versions of RFView for MAK ONE.

The directory looks like *shared-data-root/data-compatibility-version/stem/data-folders*, where:

- *shared-data-root* is the location of the shared data directory. The default is *C:/MAK/SharedData* on Windows and *~/MAK/SharedData* on Linux.
- *data-compatibility-version* is a number, such as 16 or 18.
- *stem* is the directory holding the data files, for example, *latest* for the most recent installation of that compatibility version or the date of a previous installation, such as *2024-03-14*.
- *data-folders* hold the various files that make up the MAK data files (models, audio files, fonts, and so on).

Advanced Windows users may want to install MAK Data Packages in a very specific way, for example, installing the data on a shared drive and configuring any machines with MAK ONE applications to look there for the data. Refer to the *MAK ONE Installation Guide* for instructions.

Installing the makData Package

► To install the data:

1. Install the MAK ONE Software Manager for your operating system.
2. Run the MakOneSoftwareManager executable in the `.\bin64` folder.

On Windows 10 and 11, you can also open the Start menu and then choose **MAK Technologies > MAK ONE Software Manager**. (To launch MAK ONE products more easily than the Windows 10 Start menu, please see the *MAK ONE Installation Guide* or this blog entry: [Tech Tip: Starting MAK ONE Applications on Windows 10.](#))

3. Click Select and Install Data Packages in the welcome page.
4. Specify the directory where you saved the data files. The default is your Downloads folder, but you can click the folder icon to navigate to another folder.
5. Select the data files that you want to install.



MAK Technologies recommends that you install all of the data files if you have the space and are not absolutely sure which files you require.

You must install the Core data for RFView for MAK ONE to function. To have access to other terrains and to see anything other than notional models, install the other data packages provided by MAK. If you do not install any other data, it is assumed that you are providing your own models and terrains.

6. Optionally, choose the location where you want to install the data.

MAK recommends that you install the files in the default location. While MAK ONE products allow for diverse installations, the typical use is the standard installation. If you require a more complex installation, refer to the *MAK ONE Installation Guide*.

7. If you have a previous installation of data for the same compatibility version, you are prompted to choose whether to back up the previously installed data, delete it, or overwrite it. See "Options When Installing a Data Package" below.
8. Click Install. The MAK ONE Software Manager shows the progress of the data installation.
9. When the installation is complete, click Exit.

Options When Installing a Data Package

When installing a data package using the MAK ONE Software Manager, if you have no preexisting installations of that compatibility version, it simply installs the data files in the `./SharedData/##/latest` folder (where `##` is the compatibility version).

If there is already a previous release of the same compatibility version installed, however, you are prompted to choose whether to back up, delete, or overwrite the data from the previous release.

- Backup. If you install a newer release and choose to create a backup the previous release, the files in the *latest* directory are moved to a new directory named for the data version's date in the same compatibility folder and the new files are then

installed in the latest folder. For example, the old files move to `./SharedData/18/2024-03-14` and the new files are installed in `./SharedData/18/latest`.

Creating a backup is the recommended option in most circumstances. It allows all MAK ONE applications to use the newest data, while also giving you the option to revert to a previous release using command-line options or setting a Windows environment variable. See the *MAK ONE Installation Guide* for details.

- **Delete.** The installer deletes the old data from the *latest* directory and then installs the new data.
- **Overwrite.** The installer overwrites the previous data files in the *latest* directory with the newer ones you chose to install. It also adds files that are new and ignores files that were previously installed if the package you are installing does not include that file.

1.3. The RFView for MAK ONE Directory Structure

The installation process creates the RFView for MAK ONE directory tree shown in 1.3 (RFView for MAK ONE and RFView Sidecar).

Figure 3. RFView for MAK ONE directory structure

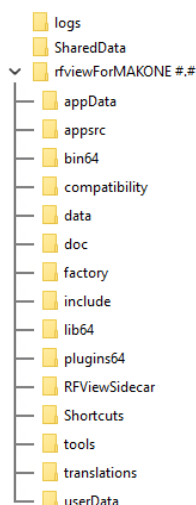


Table 1 describes the contents of the directories.

Table 1. Contents of RFView® for MAK ONE directories

Directory	Directory contents
<code>./appData</code>	Subdirectories contain data required by the RFView for MAK ONE applications, such as model definitions and saved.
<code>./appsrc</code>	Files used by the RFView Console and RFView Simple Client.
<code>./bin64</code>	Executables and dynamic link libraries.
<code>./compatibility</code>	Mapping files.
<code>./data</code>	Data specific to the function of RadarFX.

Table 1. Contents of RFView® for MAK ONE directories (continued)

Directory	Directory contents
<code>./doc</code>	RFView for MAK ONE documentation and class reference files.
<code>./factory</code>	Default settings (for reverting to “factory”).
<code>./include</code>	Header files for the RFView for MAK ONE Toolkit API.
<code>./lib64</code>	Libraries for building applications with the RFView for MAK ONE Toolkit API.
<code>./plugins64</code>	Plug-in shared libraries.
<code>./RFViewSidecar</code>	Directory for the RFView Sidecar application.
<code>./Shortcuts</code>	Contains the shortcuts to the documentation and tools, which are used for the Windows Start menu.
<code>./tools</code>	Helper applications such as the CIGI Host Emulator.
<code>./translations</code>	Translation files for localizing RFView for MAK ONE menus.
<code>./userData</code>	Terrain, scene, and image files saved by application users.

1.4. Installing and Setting Up the MAK License Manager

Before you can use a MAK ONE product, you must obtain a valid license file, install the MAK License Manager, and configure the license server and client machines. The License Manager uses a client-server architecture, so you do not need to install the License Manager on every computer on which you install MAK ONE products. You only need to install it on the computer that you will use as the license server.

 If you have already installed the License Manager for another MAK ONE product, you do not have to install it again. You only need licenses for your newly installed products.

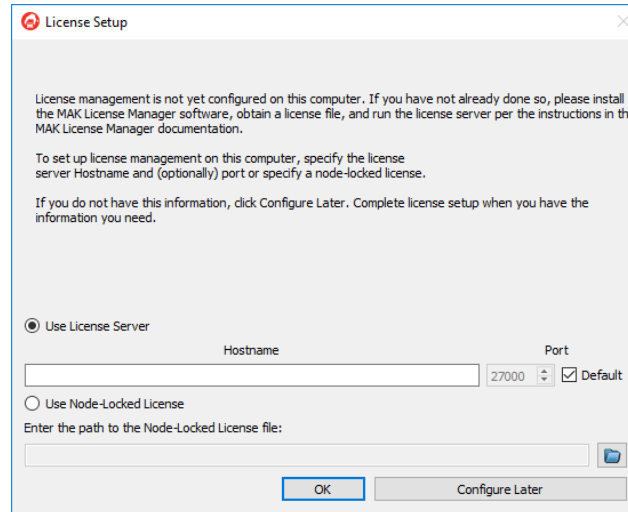
The License Manager installer is included with MAK installation media. It is separate from the product installers. You can download the installers from the [MAK ONE Product Downloads](https://www.mak.com/mak-one-downloads) page at <https://www.mak.com/mak-one-downloads>.

Complete installation and configuration instructions are included with the License Manager installer. The *MAK ONE Installation Guide* is available at <https://mak.com/mak-one-installation-guide>.

Some customers use dongle licenses instead of running a license server. Instructions for using dongles are also in the *MAK ONE Installation Guide*.

1.4.1 Specifying the License Server

The first time you run a MAK ONE product on a particular computer, the License Setup dialog box opens (Figure 4). It prompts you to enter the hostname of the license server and, optionally, a port number.

Figure 4. License Setup dialog box

If you do not know the hostname of the license server, click **Configure Later**. When you have the hostname, you can start the application again and complete the dialog box. You will not be able to run any MAK ONE products until you set up license management.

If you know the hostname, enter it in the **Hostname** box and then click **OK**. The application will start.

Under limited circumstances, MAK issues node-locked licenses. If you have a node-locked license, select the **Use Node Locked License** option and enter the path to the license file.



- If you are running MAK ONE products on the license server machine, it is also a client, so you must specify the license server on that machine, too.
- If you change the license server, the saved configuration will no longer be valid and the License Setup dialog box will open the next time you start a MAK ONE product.
- If you have changed the license server, you may want to clear the license cache. You can clear the saved license configuration by deleting the cache file. On Windows, it is *C:\Users\user_name\AppData\Roaming\MAK\licenses<n>.txt*. (The *AppData* directory is hidden by default.) On Linux, it is *.mak/license<n>.txt*. (There may be more than one cache file, for example, *licenses1.txt* and *licenses2.txt*.) You can also clear the cache in the RTI Assistant.

Setting the MAKLMGRD_LICENSE_FILE Environment Variable

As an alternative to using the license setup procedure described in the previous section, you can configure the license server or node-locked license in an environment variable. The MAKLMGRD_LICENSE_FILE environment variable identifies the server machine. If you set this environment variable, it overrides the settings stored by the license setup procedure.

For license servers, the syntax for the environment variable is: *@server_name*. For example, if the server machine is oak, set the environment variable to *@oak*.

The following sections explain how to set environment variables on the different platforms that MAK ONE products run on.

Windows

► To set MAKLMGRD_LICENSE_FILE in Windows:

1. On the Start menu, click the Settings button (⚙️). The Settings window opens.
2. In the Search box, enter `environment`. A list of choices displays.
3. Select Edit the System Environment Variables. The System Properties dialog box opens to the Advanced tab.
4. Click the Environment Variables button. The Environment Variables dialog box opens.
5. Below the System Variables list, click New. The New System Variable dialog box opens.
6. In the Variable Name field, enter `MAKLMGRD_LICENSE_FILE`.
7. Click OK to save out of each dialog box and set the variable.

Linux

On Linux, you set environment variables in your `.cshrc` (or equivalent startup file).

► To specify a license server, set the variable similarly to:

```
setenv MAKLMGRD_LICENSE_FILE @oak
```

If you are using the `sh` or `bash` shells, you set environment variables in your `.profile` file (or `.bashrc`). Set the variable similarly to:

```
MAKLMGRD_LICENSE_FILE=@oak
export MAKLMGRD_LICENSE_FILE
```

Do not put spaces around the equal (=) sign.

You are ready to run the license server and use your new licenses or MAK ONE products.

1.5. Installing an RTI

An RTI is a software library (and perhaps supporting executables) that implements an HLA Interface Specification. In HLA, applications exchange FOM data through RTI calls, which means that all HLA applications need to use an RTI. MAK Technologies has the MAK RTI, but MAK ONE products do not require that you use the MAK RTI.

i Because of differences in the low-level network mechanisms used by different RTI implementations (which include, but are not limited to, packet layout), applications that want to interoperate in the same federation execution must use the same RTI implementation.

Because RTIs are usually provided as dynamic libraries that implement a fixed API, a federation can often switch from one RTI implementation to another between runs (without even recompiling the applications), but during each run, all participants must agree on which RTI to use, much as they must also agree on which FOM to use.

For the most recent information about the RTI versions supported by MAK ONE products, please visit the Product Specifications & Dependencies page (<https://www.mak.com/mak-one-specs>).

The following instructions explain how to install the MAK RTI. If you are using an RTI provided by another vendor, please refer to that RTI's documentation for installation and setup instructions.

1.5.1 Installing the MAK RTI

To install the MAK RTI, follow the instructions in Chapter 2 of *MAK RTI Users Guide*. To install another RTI, see the vendor's documentation.

Configuring Your System to Use the MAK RTI

The RTI dynamic libraries must be located somewhere on your dynamic library search path. The path is specified in the PATH environment variable. The path is indicated by an environment variable:

- Linux — LD_LIBRARY_PATH.
- Windows — PATH.
- The MAK RTI needs to know where to find these configuration files:
 - The federation configuration file (*.fed*, *.fdd*, or *.xml*) for your federation execution (required).
 - The RID file (*rid.mtl*) (optional).

Put the configuration files in the directory from which you are running, or set the environment variable RTI_CONFIG to the directory that contains them.

Running Applications with the MAK RTI

► To run a MAK ONE product with the MAK RTI:

1. Ensure that the license server is running.
2. Start the application. The RTI Assistant prompts you to choose an RTI configuration.
3. Choose a configuration. If necessary start the rtiexec.
4. Click Connect. The application should run.

In many cases, you do not need to run the rtiexec to use the MAK RTI, however you can run the rtiexec if you want to. (It is required to use certain features of the MAK RTI.) For more information, please see your RTI documentation.

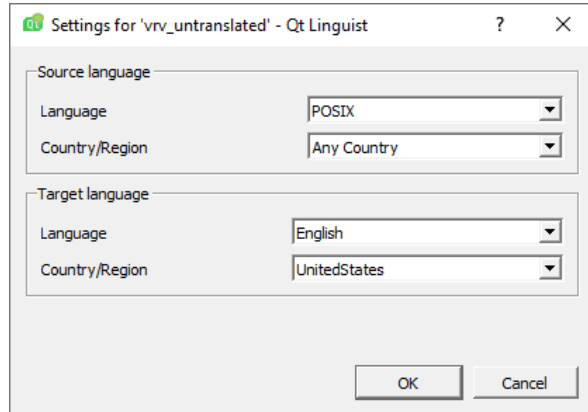
1.6. Localizing the Graphical User Interface

You can localize your MAK ONE product's graphical user interface so that it uses languages other than English. Your product includes a file called *./translations/product_untranslated.ts*, for example, . The file contains entries for all of the application-specific menu and dialog box text. The Qt Linguist tool, which is included with RFView for MAK ONE, lets you map the English text provided by MAK to your chosen language.

► To edit a resource file:

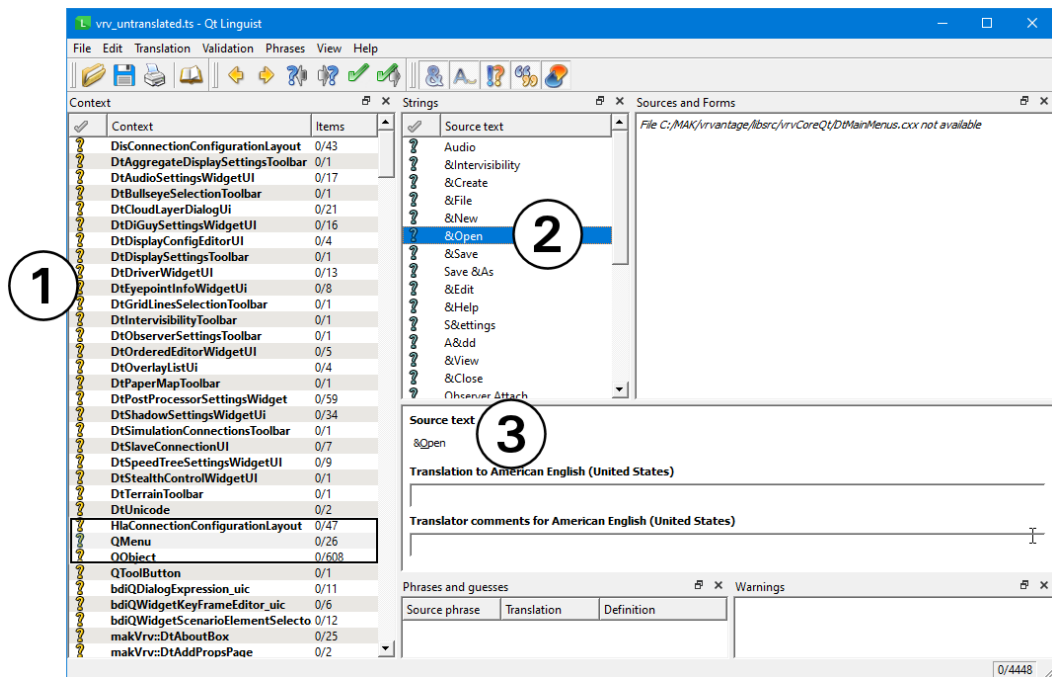
1. Start the Qt Linguist utility. It is `./translations/linguist.exe`
2. In the Qt Linguist window, choose **File > Open**. Select the file `./translations/product_untranslated.ts`. The Settings for *filename* dialog box opens (Figure 5).

Figure 5. Linguist settings dialog box



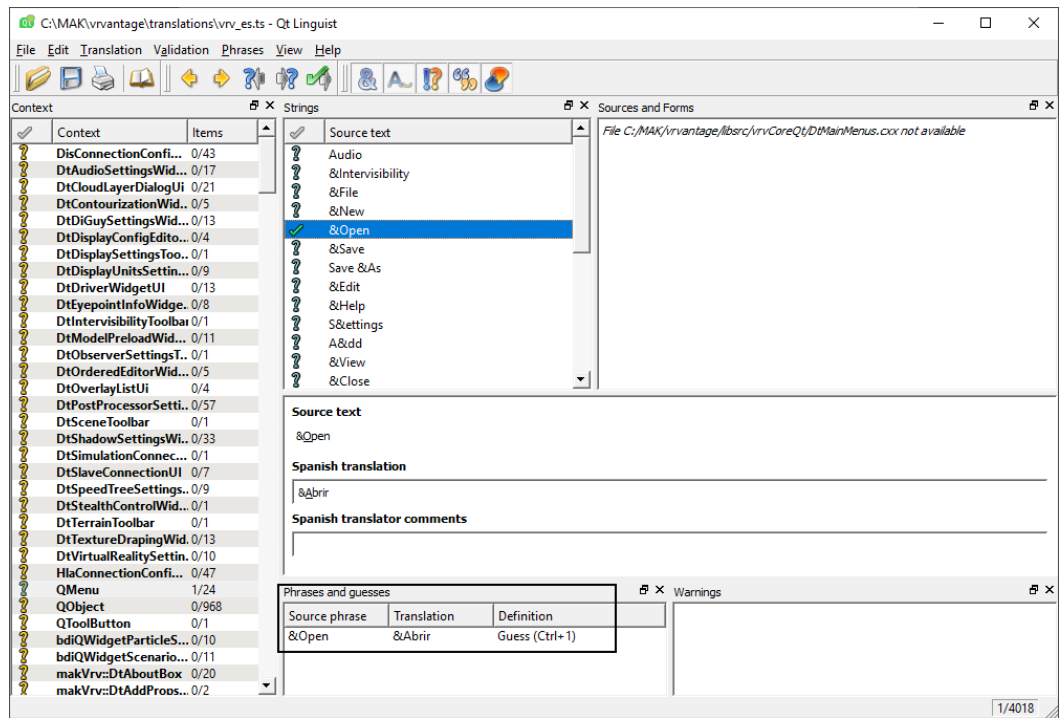
3. Select values for the source language, target language, and target region or country.
4. Click OK. The translation file opens and a list of classes displays on the left side of the window (callout 1 in Figure 6).

Figure 6. Linguist window



5. To translate text, select each class in turn and translate the text associated with that class. You do not have to understand anything about what the classes mean; you only need to translate the text. In Figure 6, **QMenu** is selected. The Strings window lists text that can be translated (callout 2). Text that has not been translated has a blue question mark in the checkmark column.
6. In the Source Text list, select the word you want to translate. The text displays under Source Text in the translation area (callout 3 in Figure 6).
7. Place the cursor in the text box below *language* Translation, where *language* is the target language that you selected when you opened the translation file.
8. Type the translation text.
9. When you are done and you are certain of the translation, select the Done and Next button () on the toolbar. A green check mark replaces the question mark (Figure 7). If you move to a different text string without clicking Done and Next, a yellow question mark is displayed for the string you translated. When you translate all elements for a class, a green check mark replaces the question mark next to the class name.

Figure 7. Completed translation for Open



10. Repeat this process for each element of source text for each class.
11. Choose **File > Release**. The file is saved in the format *filename.qm*.
12. Rename the file to a name that indicates the language to which you have translated the file, using the two-character abbreviation typically used with your language. For example, a Spanish translation would be renamed *product_es.qm*.
13. Save your translated *.ts* file.

1.6.1 Translating Other Interface Files

The file contains all the text used in the RFView for MAK ONE interface. However, there may be other text and messages that are not specific to RFView for MAK ONE that are generated by the software that implements the RFView for MAK ONE GUI. To translate this text, you need to translate the file *qt_untranslated.ts*. Follow the same procedure as you did for .

1.6.2 Applying the Language Files

► To put the language file into effect, use one of the following methods:

- Set your computer's environment to the proper language (LANG environment variable).
- Set the locale to the proper language.
- Start RFView for MAK ONE with the *-G locale* command-line option, where *locale* is the two-letter country code for the local language.



2. Using RFView® for MAK ONE Server

This chapter explains how to start and configure RFView for MAK ONE.

2.1. Starting RFView for MAK ONE Server	32
2.2. Connecting to a Simulation	32

2.1. Starting RFView for MAK ONE Server

To use RFView for MAK ONE, you run the terrain and SAR servers (VR-Vantage and RFView Sidecar, respectively) and the client application. The client could be embedded in any third-party application.

► **To start RFView for MAK ONE:**

1. On the Start menu, choose **MAK Technologies > RFView for MAK ONE 1.0**. VR-Vantage launches with the RFView Sidecar plug-in in the foreground and loads the MAK Earth (online) terrain. RFView Sidecar itself launches minimized in the background.

 RFView Sidecar is minimized because you do not need to change the parameters to use RFView for MAK ONE. If you are interested in making changes, consult ISL's RFView documentation that is included in the RFView for MAK ONE installation. For example, you can customize the settings to produce the desired polarimetric SAR image.

2. Optionally, connect to a simulation. See "2.2. Connecting to a Simulation" below.

2.2. Connecting to a Simulation

You do not need to connect to a simulation to use a client with the RFView for MAK ONE server. If you do not connect to a simulation, however, you cannot view any entities in the server.

To connect to a simulation, follow the instructions in the section "Connecting to a Simulation Network" in the *VR-Vantage User Guide*.



3. Using the RFView® for MAK ONE Simple Client

This chapter explains how to use the RFView® for MAK ONE Simple Client that demonstrates how you can request images from the RFView® for MAK ONE SAR server.

3.1. Starting RFView® Simple Client	34
3.2. Requesting a SAR Image	34
3.2.1 Specifying the Location of the Sensor	35
3.2.2 Specifying the Area of Interest	36
3.2.3 Sending the Request	36
3.3. Configuring Image Properties	36

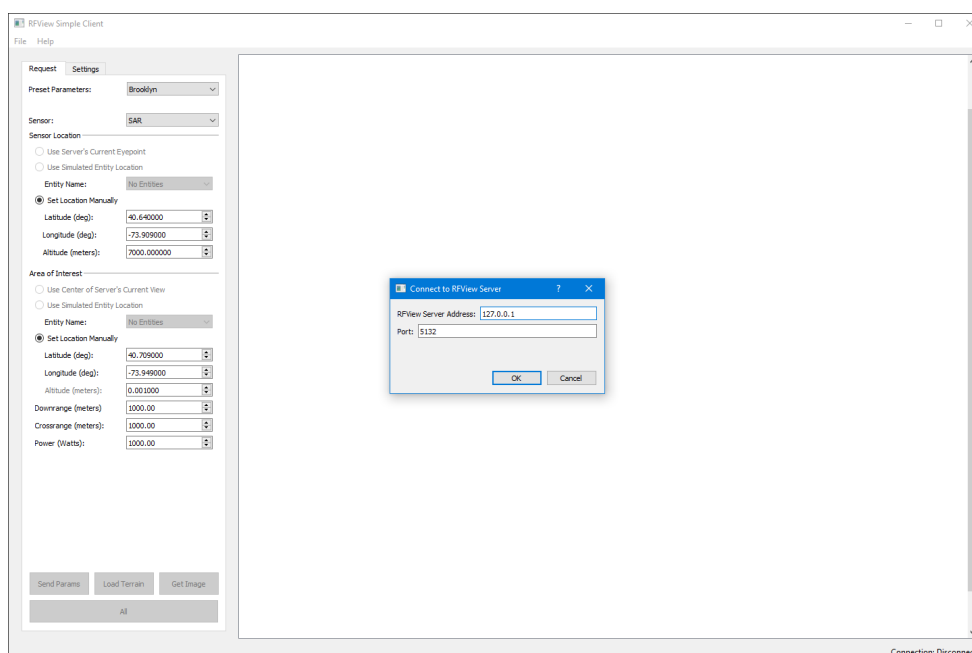
3.1. Starting RFView® Simple Client

The typical use of RFView for MAK ONE is to embed its functionality into a third-party client application. The product comes with a client application that you can use to request images from the server. This simple client is only intended to demonstrate how that can function; it is not designed for distribution to end users.

► **To start the simple client:**

1. Be sure that the RFView for MAK ONE server is running.
2. On the Start menu, choose **MAK Technologies > RFView Simple Client**. RFView Simple Client opens and the Connect to RFView Server dialog box opens with default values (Figure 8).
3. Click OK to accept the default values and connect to the RFView for MAK ONE server. RFView Simple Client is ready to use.

Figure 8. RFView Simple Client



3.2. Requesting a SAR Image

Use the client to request SAR images based on the eyepoint in the server, the location of a simulated entity, or a specific latitude, longitude, and altitude. The image includes the geometry shown in VR-Vantage at the time of capture. If there are any entities in the scene, those are included as well. (All entities include in the MAK ONE data package are materially classified.)

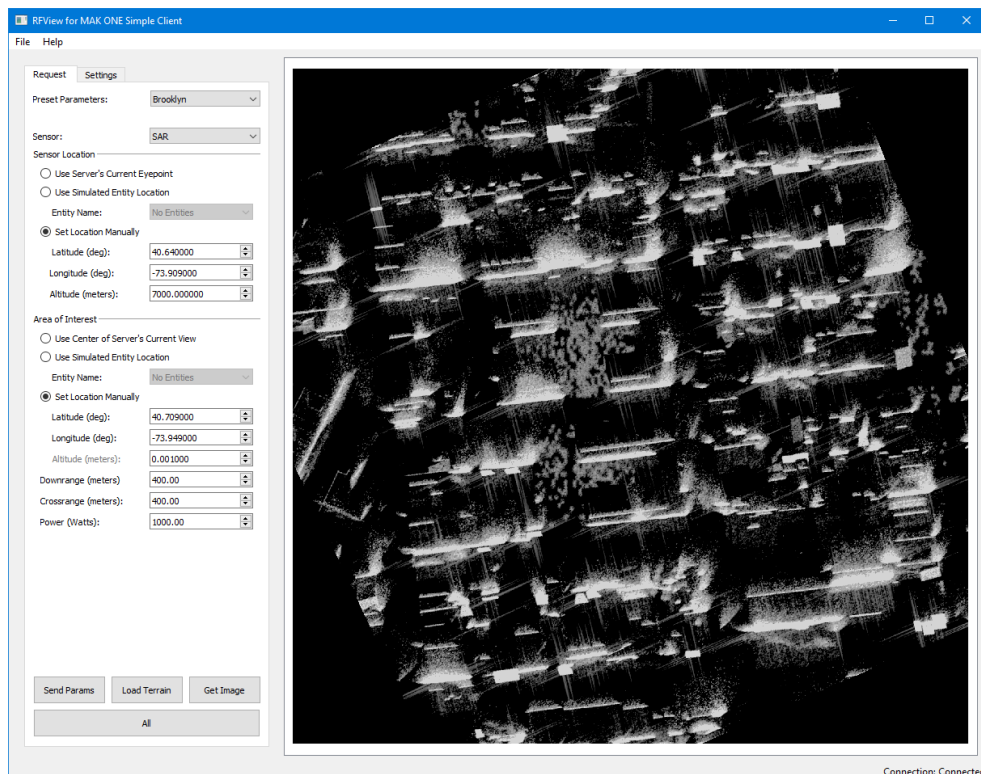
► **To request an image:**

1. On the Request tab, select an option from the Preset Parameters list. This populates the SensorLocation and Area of Interest parameters with the preset's values.
2. Optionally, refine the sensor to use by specifying the Sensor Location. For details, see "3.2.1 Specifying the Location of the Sensor" below.
3. Optionally, specify the Area of Interest. For details, see "3.2.2 Specifying the Area of Interest" on the next page.
4. Specify the down range and cross range values. For details, see "3.2.2 Specifying the Area of Interest" on the next page.
5. Optionally, change the image settings. For details, see "3.3. Configuring Image Properties" on the next page.
6. Request the image. For details, see "3.2.3 Sending the Request" on the next page.

The status area at the bottom left of the client window indicates that you have sent a request and it is loading the image. This process can take a minute or so. As the RFView Simple Client builds the SAR image, the VR-Vantage window displays the area that you selected from the Preset Parameters list.

When the full image is returned, it displays in the client window (Figure 9).

Figure 9. RFView Simple Client with image of Brooklyn



3.2.1 Specifying the Location of the Sensor

Select an option for the specifying which sensor will capture the image:

- Use Server's Current Eyepoint. This option is most useful when both the server and the client are running on the same machine and you can manually move the observer into position and then request a radar image. The eyepoint should be above the target location.
- Use Simulated Entity Location. If the server is connected to a simulation, you can select an entity that is in the scene. The eyepoint should be above the target entity.
- Set Location Manually. Enter a location manually in latitude, longitude, and altitude.

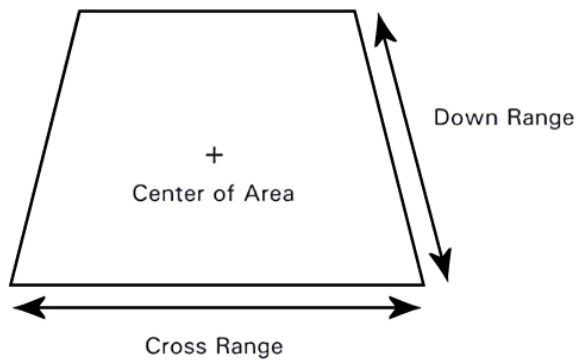
3.2.2 Specifying the Area of Interest

Select an option to specify a particular area of interest at the preset location you selected.

- Use Center of Server's Current View. An area centered on the server's eyepoint.
- Use Simulated Entity Location. An area centered on the selected simulated entity.
- Set Location Manually. An area you specify by latitude, longitude, and altitude.

The down range and cross range values specify the size, in meters, of the area of interest centered on the target location, as illustrated in Figure 10.

Figure 10. Area of interest



3.2.3 Sending the Request

- The Send Params button send the parameters of where the sensor is and where it is looking. It does not re-render the image.
- The Load Terrain button loads the terrain at the specified location. It does not render the image.
- The Get Image button gets the SAR image at the eyepoint.
- The All button runs all of these processes.

3.3. Configuring Image Properties

You can specify the properties of the images that the server returns. You can specify these image properties:

- Size. The size of the image in pixels.
- Rotation. The image can be rotated so that North is up, the sensor's view (track) is up, or its line of sight (LOS) is up.



Index

A

area of interest, 36

C

callstack.log file, 18

configuration

image, 36

MAK RTI file, 26

crash, application, 20

cross range, 36

D

data

install, 17

options, 21

debugging

callstack.log file, 18

dmp file, 18

file location, 20

directory

Windows default install, 18

dmp file

debugging, 18

location, 20

down range, 36

E

entity, viewing, 32

environment variable

MAK_INSTALL_DIR, 18

MAK_LOG_DIR, 20

MAKLMGRD_LICENSE_FILE, 24

RTI_CONFIG, 26

executable

installation location

Windows, 18

F

FDD file, 26

FED file, 26

file

FDD, 26

FED, 26

RID, 26

FLEXIm, 23

FOM, 25

G

GUI

localizing, 26

H

hostname
 license server, 23

I

image
 orientation, 36
 requesting, 34
 resolution, 36
installation, 17
 data
 options, 21
 default directory, Windows, 18
 RTI, 26
 Windows, 17
interest area, 36

L

license manager, 23
license server
 hostname, 23
localization
 RFView for MAK ONE GUI, 26
log file
 callstack, 18
 location, 20

M

MAK data
 install
 options, 21
MAK RTI configuration, 26
MAK_INSTALL_DIR environment variable, 18
MAKLMGRD_LICENSE_FILE environment variable,
 24

P

path
 default Windows install, 18
 MAK RTI, 26
PDB files
 Linux, 19
 Windows, 19

product installation, 17
 Windows, 17

Q

Qt Linguist, 26

R

region, 36
RID file, 26
rid.mtl file, 26
RTI, 25
RTI_CONFIG environment variable, 26

S

SAR image, 34
sensor
 location, 35
server, 32
sidecar, 32
simulation, connecting, 32

T

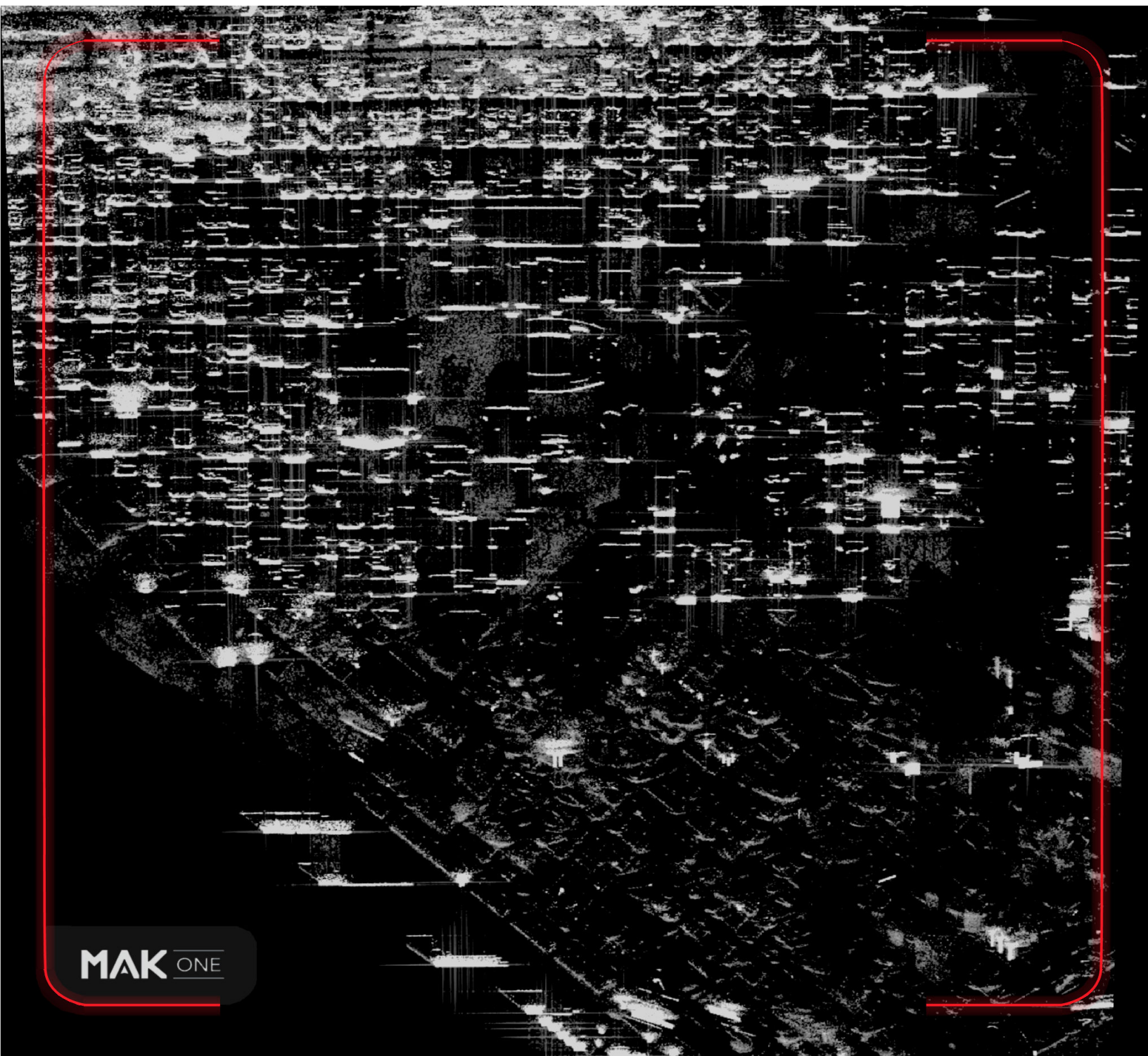
translation, RFView for MAK ONE GUI text, 26

W

Windows
 default install directory, 18
 product installation, 17

X

XML
 RTI configuration file, 26



MAK ONE

RFV-1.0-02-250408