

# **DODS Quick start**

---

## **DODS Quick start**

Together Teamlösungen EDV-Dienstleistungen GmbH

Elmargasse 2-4

A-1190

Vienna

Austria

+43 (0) 5 04 04 - 122

+43 (0) 5 04 04 - 11 122

<office@together.at>

<http://www.together.at/together/index.html>

Copyright © 2006 Together Teamlösungen EDV-Dienstleistungen GmbH

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, photocopying, recording or otherwise, without the prior written permission of the Together Teamlösungen EDV-Dienstleistungen GmbH.

Together Teamlösungen EDV-Dienstleistungen GmbH DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

---

---

## Table of Contents

1. Introduction .....	1
2. System requirements .....	2
3. DODS source building .....	3
4. Starting DODS generator .....	5
DODS Generator Wizard .....	5
5. Running DODS in Ant .....	8

---

# Chapter 1. Introduction

This document describes features needed for building, getting up and running quickly with DODS:

- System requirements for DODS
- How to build DODS from source
- How to start DODS generator wizard
- How to use DODS Ant task

---

# Chapter 2. System requirements

The minimum recommended system configuration is:

- 128 MB RAM
- 200 MHz processor or better
- 50 MB disk space (see note below)
- Java Development Kit (JDK) 1.4
- JDBC-compliant SQL database for database access

Note: The disk space specified above reflects the minimum system requirements to install and run the DODS example applications. A minimum installation consists of:

- JDK 1.4.1 (approximately 50 MB)
- DODS 6.0 (approximately 30 MB)
- InstantDB (approximately 2 MB)

---

## Chapter 3. DODS source building

In Enhydra5.1(6) is included binary version of DODS (without source). So, DODS can not be build out of source in Enhydra, only as independent project.

To build independent DODS, it is necessary to do the following actions:

- Unix stile slashes (/) must always be used instead of DOS stile backslashes (\).
- Start Command Prompt and go to <DODS\_SOURCE> directory.
- To configure DODS you can call configure batch file with following options:

```
configure
  [-version version_number] [-release release_tag] [-jdkhome jdk_home_dir]
  [-debug on/off] [-optimize on/off] [-instldir installdir]
```

where:

- -version sets version\_number. Default: 0.5.
- -release sets release\_tag. Default: 1.
- -jdkhome sets java jdk\_home\_dir. Default: Path to system registred (if any) jdk.
- -debug compiles source with debug information (on/off). Default: off.
- -optimize sets whether the source should be compiled with optimization or not (on/off). Default: on.
- -instldir the path to your installation directory (see "Make Options" --> make install)

Configure without parameters sets configuration parameters to default values.

- To build DODS start Command Prompt and go to <DODS\_SOURCE> directory.

DODS building is completely Ant based. You can give one of the following options to the make command:

- make - builds and configures DODS with javadoc and docbook documentation
- make buildAll - builds and configures DODS with javadoc and docbook documentation
- make buildOptimize - builds, optimizes and configures DODS with javadoc and docbook documentation
- make buildNoDoc - builds and configures DODS without documentation building
- make install - copies and configures DODS without source compiling
- make distributions - builds and configures DODS with javadoc and docbook documentation and creates distribution; nsis 2.0b should be included in DODS if doesn't exist (files makensis.exe and makensisw.exe in Dods/Install/Windows/install directory)
- make optimizeDistributions - builds and configures DODS with javadoc and docbook documentation and creates optimized distribution; nsis 2.0b should be included in DODS if doesn't exist (files makensis.exe and makensisw.exe in Dods/Install/Windows/install directory)
- make clean - removes the output folder (in order to start a new compilation from the scratch)

- make help - displays all options

where <DODS\_HOME> is directory in which DODS is built.

- After DODS building, you MUST add <DODS\_HOME>\bin directory to the beginning of the system PATH.

---

# Chapter 4. Starting DODS generator

If you are running on Windows, to start DODS generator, you should use batch files in

- <DODS\_HOME>/bin directory, for independent DODS, or
- <ENHYDRA\_HOME>/bin directory, for DODS within Enhydra 5.1 or 6.0

where <ENHYDRA\_HOME> is home (root) directory of Enhydra 5.1 or 6.0

At the command window, change directory to the /bin directory.

For example,

- for independent DODS:

```
cd
    /usr/local/<DODS_HOME>/bin
```

if you installed DODS in /usr/local/<DODS\_HOME> directory

- for DODS within Enhydra 5.1 or 6.0:

```
cd
    /usr/local/<ENHYDRA_HOME>/bin
```

if you installed Enhydra 5.1 or 6.0 in /usr/local/<ENHYDRA\_HOME> directory

The dods generator start batch file is located in this bin directory. So, to start the dods generator, enter:

```
dods
```

If you are running Linux, you can use start scripts in the same directory. So to start the dods generator on Linux, enter:

```
./dods
```

- Note:

<DODS\_HOME>/bin directory should be added in system path. Then, DODS can be started from any directory (by typing dods).

You will see DODS Generator Wizard displayed in the output window.

## DODS Generator Wizard

Figure 1: DODS Generator Wizard

- You must choose project root and path to the doml file.
- Config directory field contains path to custom configuration folder (which contains dodsConf.xml file). It is used to generate java source code and SQL scripts. If the path is set to any other path than default (offered), in the application's configuration file should be set parameter

```
DatabaseManager.ConfigurationDir
```

to new path of the custom configuration folder.

- There are four options on the Generator Wizard:
  - SQL generate
    - SQL Splitter
  - Java generate
    - Compile Java

At least one of the Generate fields must be checked.

- There are two combo boxes on the Generator Wizard. Template set combo box contains possible template sets:
  - standard
  - <user\_defined\_templates>
- DB vendor combo box contains list of database vendors. If one of these vendors is selected, this database will overwrite database declared in DOML file.
- There is a possibility on the Generator Wizard for generating the following types of documentation:
  - HTML
  - PDF
  - XMI
  - PTL
- On the Generator Wizard, there is also a check box:

`overwrite`

for code generating (java and sql), no matter if the code already existed.

You can manually start dods generator without using wizard by typing dods with additional parameters:

Command line:

```
dods [-?/help] [-a action] [-t templateset] [-b/-database] [-c confPath]
[-f/force] [-h/html] [-p/pdf] [-x/xmi] [-r/ptl] domlfile outputdir
```

where:

- outputdir is full path to output directory that will be used.
- domlfile is full path to .doml file for generating code.

options:

- [-? -help] shows help.
- [-a action] - ant task parameter for code generation:

- dods:build\_all - to create all sql files and java classes (default).
- dods:sql - to create only sql files.
- dods:java - to create only java files and to compile them.
- dods:javaNoCompile - to create only java files and not to compile them.
- dods:noCompile - to create SQL files and java files and not to compile them.
- dods:build\_all\_split - to create all sql files and java classes and to compile them. SQL files will be divided into separate files using SQLSplitter .
- dods:sqlsplit - to create only sql files and separate them in different files using SQLSplitter.
- dods:noCompileSplit - to create SQL files and separate sql commands using SQLSplitter and java files and not to compile them.
- dods:generatorOff - to disable generating and compiling of java source code, for generating documentation only (you stil need to set documentation property: html, pdf, ptl, xmi).
- [-t templateset] - template set for generating java and sql code:
  - standard - generate standard java code (default).
  - <user defined> - any user defined template set.
- [-b/-database] - sets database vendor for generating sql
- [-c confPath] - sets folder with dodsConf.xml file
- [-f/-force] - with this switch, code will be always generated, without it, only changes will be regenerated.
- [-h/-html] - generates DODS html documentation from .doml file.
- [-p/-pdf] - generates DODS pdf documentation from .doml file.
- [-x/-xmi] - generates DODS xmi documentation from .doml file.
- [-r/-ptl] - generates DODS ptl (Rational Rose) documentation from .doml file.

---

# Chapter 5. Running DODS in Ant

Invokes DODS to generate a set of java classes from a DOML file. The files will only be regenerated/compiled if the date on the DOML file is newer than at least one of the generated files.

This taskdef extends Ant's <javac> task; refer to documentation for parameters that affect compilation.

Typically made visible to an Ant build file with the following declaration:

```
<taskdef name="dods" classname="org.enhydra.ant.taskdefs.Dods"/>
```

Parameters:

domlfile - The doml input file describing data object mapping. Required = Yes.

outputDir - Target for generated classes, expressed as a directory path. Required = Yes.

force - Forces DODS always to regenerate source files. Possible values: ("true", "false"(default)). Required = No.

action - Name of Ant task from generate.xml. Required = No.

templateDir - Name of folder for template set for generating java code, expressed as a directory path. Required = No.

templateSet - Template set for generating java code. Required = No.

confDir - Path to custom configuration folder (If the path is set to any other path than default (offered), in the application's configuration file should be set parameter:

```
DatabaseManager.ConfigurationDir
```

to new path of the custom configuration folder). Required = No.

database - Sets database vendor for generating sql. Required = No.

html - Indicates DODS to generate html documentation from .doml file. Possible values: ("true", "false"(default)). Required = No.

pdf - Indicates DODS to generate pdf documentation from .doml file. Possible values: ("true", "false"(default)). Required = No.

xmi - Indicates DODS to generate xmi documentation from .doml file. Possible values: ("true", "false"(default)). Required = No.

ptl - Indicates DODS to generate ptl (Rational Rose) documentation from .doml file. Possible values: ("true", "false"(default)). Required = No.

action parameters:

- without parameters - to create all sql files and java classes and to compile them.
- dods:build\_all - to create all sql files and java classes.
- dods:sql - to create only sql files.
- dods:java -to create only java files and to compile them.

- dods:javaNoCompile -To create only java files and not to compile them.
- dods:noCompile -To create SQL files and java files and not to compile them.
- dods:build\_all\_split - to create all sql files and java classes and to compile them. SQL files will be divided into separate files using SQLSplitter.
- dods:sqlsplit - to create only sql files and separate them in different files using SQLSplitter.
- dods:noCompileSplit - To create SQL files and separate sql commands using SQLSplitter and java files and not to compile them.
- dods:generatorOff - to disable generating and compiling of java source code, for generating documentation only (you stil need to set documentation property: html, pdf, ptl, xmi).

templateset parameters:

- standard - generate standard java code.
- <user defined> - any user defined template set.

Example:

```
<dods doml="${basedir}/discRack.doml"
    outputDir="${basedir}/src"
    templateSet="standard"/>
```