

IKAN ALM

Using Phase Parameters

Release 1.0

Februari 2019



IKAN Development N.V.
Kardinaal Mercierplein 2
2800 Mechelen
BELGIUM

© 2006 - 2019 IKAN Development N.V.

No part of this document may be reproduced or transmitted in any form or by any means, electronically or mechanically, for any purpose, without the express written permission of IKAN Development N.V.

The IKAN Development and IKAN ALM logos and names and all other IKAN product or service names are trademarks of IKAN Development N.V.
All other trademarks are property of their respective owners.

Table of Contents

Chapter 1 - Purpose	1
1.1. IKAN ALM Phases.....	1
1.2. IKAN ALM Setup - example	1
Chapter 2 - Step-by Step Overview	3
Chapter 3 - Introduction	4
3.1. General Definition of the Phase	4
3.2. Modifying the Default Initial Values in IKAN ALM	4
3.3. Refining the Parameter Values on Different Levels.....	5
3.4. Order of Precedence	6
3.5. Influencing the Order of Precedence	7
3.6. Best Practices.....	7
3.7. Parameter characteristics	7
3.8. Example UpdateDB Phase: Parameter "rdbms.type"	8
Chapter 4 - Initial Parameter Values	10
4.1. Overview.....	10
4.2. Modifying the Initial Parameter Values	10
Chapter 5 - Environment Phase Parameters	12
5.1. The Environment Phases Overview	12
5.2. Modifying the Environment Phase Parameter Value.....	14
Chapter 6 - Machine Parameters	17
6.1. Creating a Machine Parameter.....	17
6.2. Replacing the Value of the Environment Phase Parameter.....	19
Chapter 7 - Environment Parameters	22
7.1. Creating an Environment Parameter.....	22
7.2. Replacing the Value of the Environment Phase Parameter.....	24

Chapter 8 - Overview Connected Environment Parameters	27
Chapter 9 - Phase Logs.....	29

CHAPTER 1

Purpose

This document is intended for advanced users of IKAN ALM.
It describes how to optimize the use of the IKAN ALM Phase Parameters.

1.1. IKAN ALM Phases

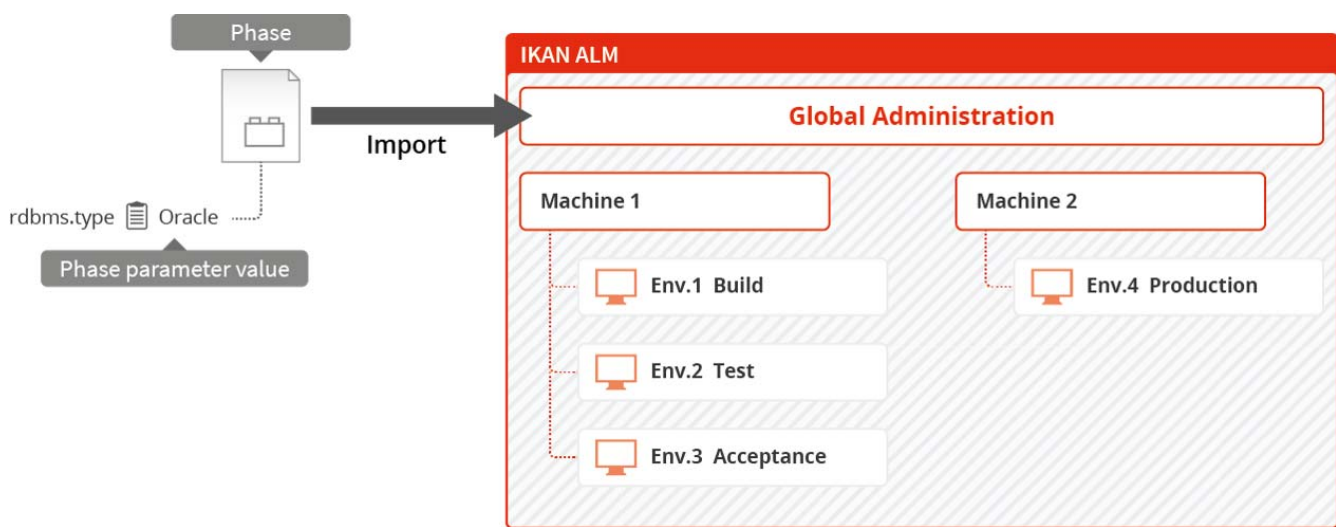
IKAN ALM Phases are predefined, parameter-driven, reusable building blocks. They represent specific tasks or actions that are executed to automate the Lifecycle Management Process.

For more general information on IKAN ALM Phases, refer to the appendix *Phases - General Information* in the *IKAN ALM User Guide* or the document *Using and Developing Phases*. Both can be found on the [IKAN ALM website](#).

1.2. IKAN ALM Setup - example

Throughout this document, we will use the following IKAN ALM setup to illustrate the different aspects of defining and refining the value of the Phase Parameters.

In our setup, we have two Machines and on each of them one or more Environments have been defined. Machine 1 is using three Environments, for Build, Test and Acceptance, and Machine 2 is being used for Production.



Each Environment uses one or more Phases.

When you import the Phase into IKAN ALM, the initial parameter values from the script are set in the Global Definition of the Phase. The first thing to do will be to verify and adapt those values to your own configuration.

Next, you link the Phase to the Build and/or Deploy Environments where you want to use the Phase. The initial values will be set as the values of the Environment Phase Parameters.

When that is done, and if required, you can refine the parameter values on Machine and Environment level. On top of that, Machine and Environment Parameter can be defined as editable and dynamic so that their value can be chosen at the moment the Phase is executed.

As the Environment Phase Parameter value is the value that will be used when creating the Level Request, you might need to override that value by the value of the Machine or Environment Parameter value.

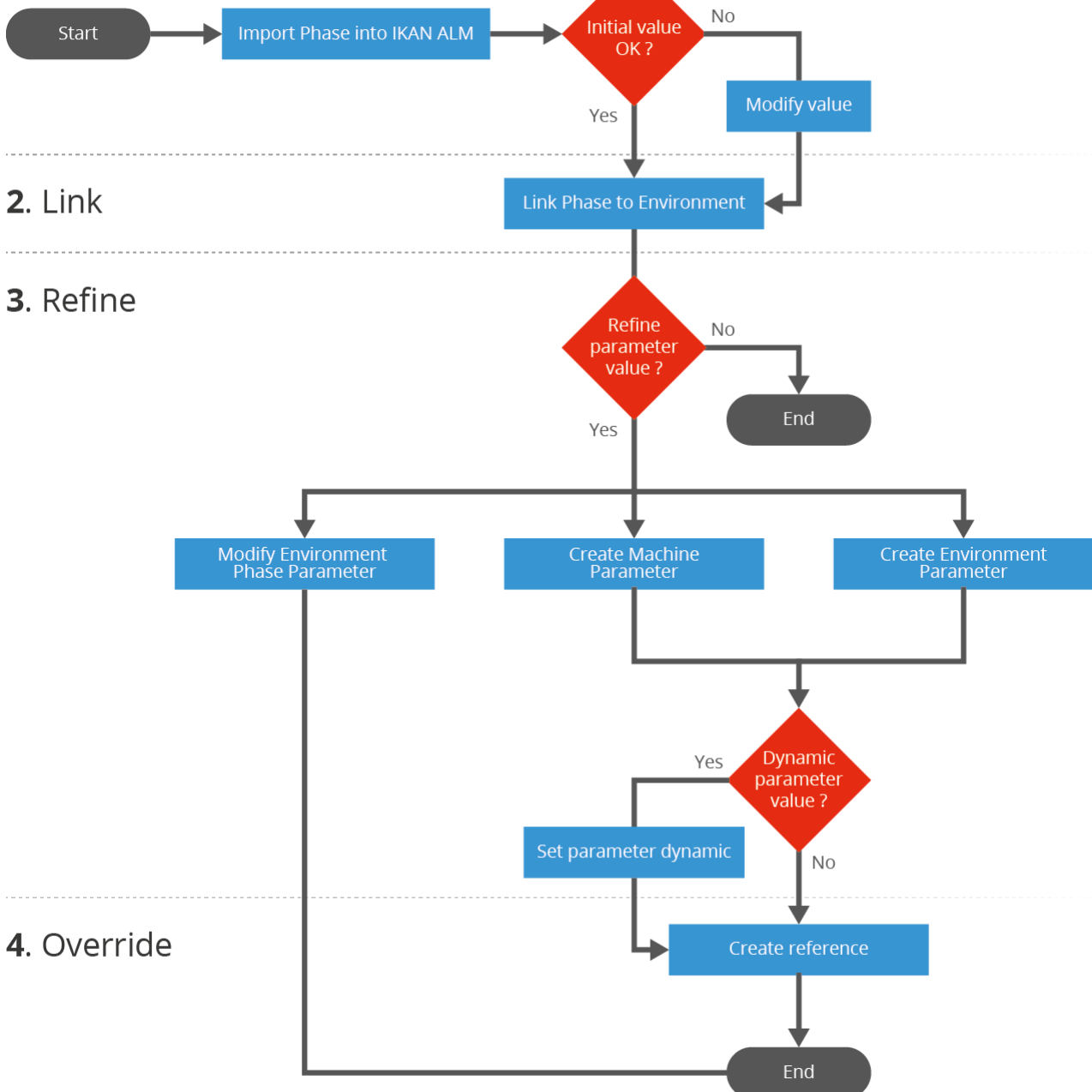
How all that is done, is explained in the remaining of this document:

- [Introduction](#) (page 4)
- [Initial Parameter Values](#) (page 10)
- [Environment Phase Parameters](#) (page 12)
- [Machine Parameters](#) (page 17)
- [Environment Parameters](#) (page 22)

Step-by Step Overview

The process of defining and refining the parameter values in IKAN ALM can be schematised as follows.

1. Import



3.1. General Definition of the Phase

All parameters used by the Phase are defined in the phase definition XML file that is included in the phase bundle.

When you import an IKAN ALM phase, all defined parameters will be listed with the following information: name, default value, description, mandatory and/or secure and integration type.

The screenshot displays the 'Global Administration > Edit Phase' window. The 'Phase Info' section shows the following details:

- Name:** com.ikanalm.phases.database.general.updateDB
- Execution Type:** ANT
- Version:** 1.0.0
- Core Phase:** No
- Default Display Name:** Update DB
- Certified:** No
- Author:** IKAN
- Released:** Yes
- Description:** Ant Phase to update a database
- Uploaded Files:** UpdateDB.xml, lib/nsqldb.jar

The 'Phase Parameters' section contains a table with the following data:

Name	Default value	Description	Mandatory	Secure	Integration Type
rdbms.type	ORACLE	The database type. One of the following: HSQLDB, MYSQL, MSSQL, ORACLE or DB2	✓		None
rdbms.server	localhost	Database host	✓		None
rdbms.port		Database port (optional)			None
rdbms.dbname	almDemo	Database name	✓		None
rdbms.dbschema		Database schema (optional)			None
rdbms.user	sa	Database user with update permissions	✓		None
rdbms.pwd	*****	Password of the database user	✓	✓	None
sql.script		Full path to the sql script to be executed			None
alm.phase.builder			✓		ANT
alm.phase.extractBundle	true		✓		None

3.2. Modifying the Default Initial Values in IKAN ALM

Before adding the imported Phase to a Build or Deploy environment, you can modify the default values at Global Administration level.

Those default values will be used to initialize the Environment Phase Parameters when you add the Phase to a Build or Deploy Environment.

3.3. Refining the Parameter Values on Different Levels

The value of the Phase Parameter can be specified at different levels.

Environment Phase Level

At the moment the Phase is linked to an Environment, the Environment Phase Parameters are set. Their value is taken from the initial value specified in the Global Definition of the Phase.

Important: The value of the Environment Phase Parameter is the value that will be used when executing the Phase (see also [Order of Precedence](#) (page 6)).

If required, you can modify this value for a particular Environment Phase. For example, suppose the default value of the Environment Phase Parameter is set to `MYSQL`, but for the Test Environment you are using `DB2`. In that case, you could modify the value of the Environment Phase Parameter.

For more information, refer to the section [Environment Phase Parameters](#) (page 12).

Note: The value of an Environment Phase Parameter is always fixed. If you want to be able to choose the value at the moment you create the Level Request, you will have to use a dynamic or editable Machine or Environment Parameter.

Machine Level

Machine Parameters are parameters defined for a specific Machine. They will be automatically available for all Environments using that Machine. This avoids having to define (Build or Deploy) Environment Parameters for each Environment linked to a Machine.

Machine Parameters can be defined as editable or dynamic. This allows the user to choose their value at the moment they create the Level Request.

For example, suppose you have several machines, all using `MYSQL` except for one machine which is using `HSQldb`. In that case it will be interesting to define a Machine Parameter for that particular machine.

For more information, refer to the section [Machine Parameters](#) (page 17).

Environment Level

Environment Parameters are parameters defined for a specific Build or Deploy Environment.

Environment Parameters can be defined as editable or dynamic. This allows the user to choose their value at the moment they create the Level Request.

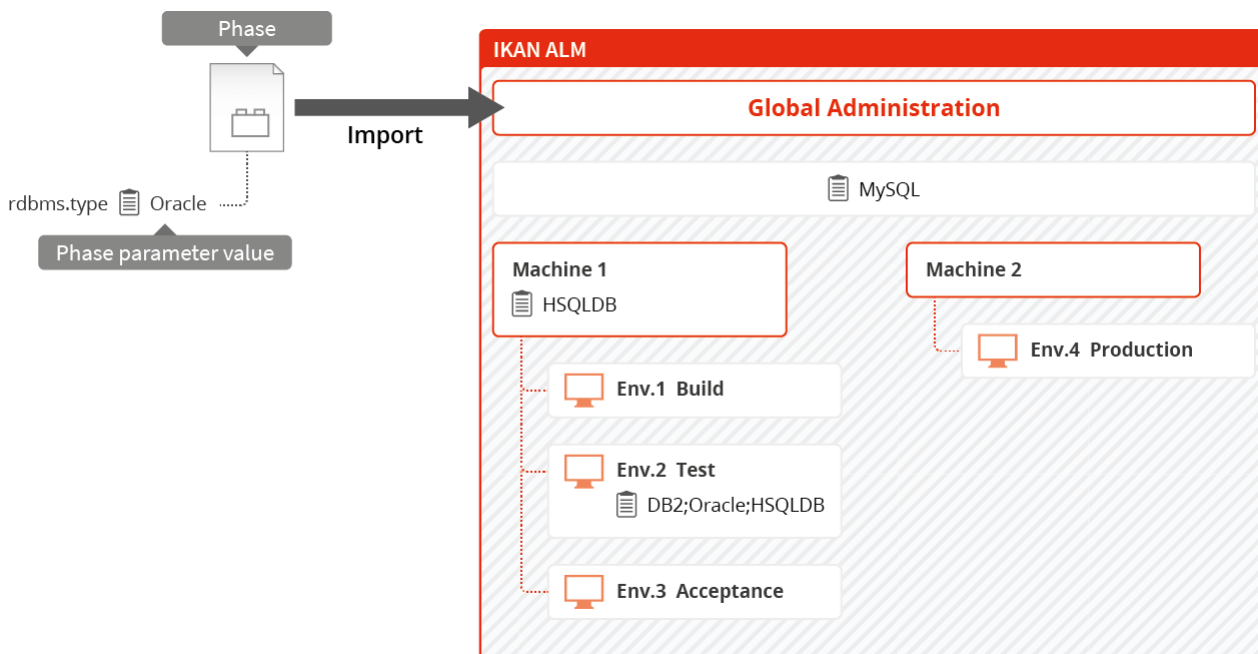
To elaborate further on our example: suppose that for one particular environment you are using a different database. If the value of that database is fixed, you can modify the value of the Environment Phase Parameter for that particular environment (as explained above) OR create an Environment Parameter. If you want to be able to choose the value at the moment you create the Level Request, you must use an Environment Parameter as the value of an Environment Phase Parameters is always fixed. In case of a dynamic parameter, the first value in the list will be the default value.

For more information, refer to the section [Environment Parameters](#) (page 22).

3.4. Order of Precedence

Very important: when processing a Phase in the context of a specific Level Request, the value of the Environment Phase Parameter is ALWAYS USED.

If the value of a Machine or Environment parameter should be used instead, you will need to replace the value of the Environment Phase Parameter by a reference to the value of the Machine or Environment Parameter that need to be used. See [Influencing the Order of Precedence](#) on page 7.



Example

The above example, shows the values for the “rdbms.type” parameter used by the UpdateDB Phase.

In the phase definition XML file, the value of the parameter is set to ORACLE.

The initial value has been modified to MYSQL before linking the Phase to the Environment, so the value of the Environment Phase Parameter is set to MYSQL.

On Machine Level, the value of the parameter is set to HSQLDB for Machine 1. On the Test environment, a dynamic Environment Parameter has been created.

As the value of the Environment Phase Parameter always takes precedence, when running a Level Request, the value MYSQL will be used.

To enforce the usage of a Machine or Environment Parameter, you have to use the solution explained in the section [Influencing the Order of Precedence](#) (page 7).

3.5. Influencing the Order of Precedence

How to proceed to make sure that instead of the value of the original Environment Phase Parameter the value of the Machine or Environment Parameter will be used?

To do so, you will need to replace the value of the Environment Phase Parameter with a reference to the value of the Machine or Environment Parameter value. The syntax to be used is: `${ReplacingParameter}`.

For more information, refer to the section [Replacing the Value of the Environment Phase Parameter](#) (page 19).

3.6. Best Practices

Unique Name

When using Ant Phases, the name of the replacing Machine or Environment Parameter must be unique, as Ant will not accept a reference to a parameter with the same name.

Naming Convention

We recommend you to use the name of the original Phase parameter preceded by the machine or environment name. That way it will be easy to track where the value used by the Level Request has been taken from, e.g. `ikanalm004v.rdbms.type`.

See also [Phase Logs](#) (page 29).

3.7. Parameter characteristics

Environment Phase Parameters always have a fixed value.

Machine and Environment Parameters on the other hand can have the following additional characteristics:

- *Editable Parameters* have a default value, but you can change this value each time you create a Level Request.
- *Dynamic Parameters* dispose of a list of allowed values. You can select one of these allowed values from the drop-down list, when you create a Level Request.

See also [Creating a Machine Parameter](#) (page 17) or [Creating an Environment Parameter](#) (page 22).

3.8. Example UpdateDB Phase: Parameter "rdbms.type"

Parameter values used when creating the Level Request **WITHOUT** using a reference to a Machine or Environment Parameter

Refinement Level	Value for the "rdbms.type" parameter at the different Levels				
General Phase Definition at import	ORACLE	ORACLE	ORACLE	ORACLE	ORACLE
Modified Initial Value (Global Admin)		MYSQL	MYSQL	MYSQL	MYSQL
Machine Parameter			HSQLDB	HSQLDB	HSQLDB
(Build/Deploy) Environment Parameter				DB2	DB2
Modified Environment Phase Parameter					MSSQL
Value you want to use when creating the Level Request	ORACLE	MYSQL	HSQLDB	DB2	MSSQL
Value that will be used when creating the Level Request (due to the Order of Precedence)	ORACLE	ORACLE	ORACLE	ORACLE	MSSQL

Parameter values used when creating the Level Request **USING** a reference to a Machine or Environment Parameter

Refinement Level	Value for the "rdbms.type" parameter at the different Levels				
General Phase Definition at import	ORACLE	ORACLE	ORACLE	ORACLE	ORACLE
Modified Initial Value (Global Admin)		MYSQL	MYSQL	MYSQL	MYSQL
Machine Parameter, e.g., machine.rdbms.type			HSQLDB	HSQLDB	HSQLDB
(Build/Deploy) Environment Parameter, e.g., environment.rdbms.tyoe				DB2	DB2
Modified Environment Phase Parameter					MSSQL
Value of the Environment Phase Parameter	ORACLE	MYSQL	\${machine.rdbms.type}	\${environment.rdbms.type}	MSSQL
Value that will be used when creating the Level Request (using references)	ORACLE	MYSQL	HSQLDB	DB2	MSSQL

For more detailed information, refer to the following sections:

- General Phase Definition at import: [Overview](#) (page 10)
- Modified Initial Value (Global Admin): [Modifying the Initial Parameter Values](#) (page 10)
- Machine Parameters: [Machine Parameters](#) (page 17)
- (Build/Deploy) Environment Parameters: [Environment Phase Parameters](#) (page 12)
- Environment Phase Parameters: [Modifying the Environment Phase Parameter Value](#) (page 14), [Replacing the Value of the Environment Phase Parameter](#) (page 19)
- Final value used when creating the Level Request: [Phase Logs](#) (page 29)

Initial Parameter Values

4.1. Overview

When you import a Phase into IKAN ALM, the initial parameter values from the script are set in the Global Definition of the Phase. Next, when you add the Phase to a Build and/or Deploy Environment, those initial values will be set as the values of the Environment Phase Parameters, which will be used when executing the Phase.


Of course, it is unlikely that all initial parameter values are correct for your configuration. So, the first thing to do will be to verify and, if required, modify the values.


This must be done BEFORE you link the Phase to a Build and/or Deploy Environment. Once the Phase has been linked to an Environment, modifying the default value will not update the value everywhere the Phase is being used.

4.2. Modifying the Initial Parameter Values

Note: To modify the initial parameter values you need Global Administrator Access Rights.

1. Select *Global Administration > Phases > Overview*.

- Click the  *Edit* icon in front of the Phase whose initial parameter values need to be modified. The *Edit Phase* screen will be displayed.

Global Administration > Edit Phase 





















Phase Info

Name	com.ikanalm.phases.database.general.updateDB	Execution Type	ANT
Version	1.0.0	Core Phase	No
Default Display Name	Update DB	Certified	No
Author	IKAN	Released	Yes
Description Ant Phase to update a database			
Uploaded Files UpdateDB.xml lib/hsqldb.jar			


[History](#) [Save](#) [Edit](#) [Refresh](#) [Overview](#)

[Export](#) [Copy](#)

Phase Parameters

	Name	Default value	Description	Mandatory	Secure	Integration Type
 	rdbms.type	ORACLE	The database type. One of the following: HSQLDB, MYSQL, MSSQL, ORACLE or DB2	<input checked="" type="checkbox"/>	<input type="checkbox"/>	None
 	rdbms.server	localhost	Database host	<input checked="" type="checkbox"/>	<input type="checkbox"/>	None
 	rdbms.port		Database port (optional)	<input type="checkbox"/>	<input type="checkbox"/>	None
 	rdbms.dbname	almDemo	Database name	<input checked="" type="checkbox"/>	<input type="checkbox"/>	None
 	rdbms.dbschema		Database schema (optional)	<input type="checkbox"/>	<input type="checkbox"/>	None
 	rdbms.user	sa	Database user with update permissions	<input checked="" type="checkbox"/>	<input type="checkbox"/>	None
 	rdbms.pwd	*****	Password of the database user	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	None
 	sql.script		Full path to the sql script to be executed	<input checked="" type="checkbox"/>	<input type="checkbox"/>	None
 	alm.phase.builder			<input checked="" type="checkbox"/>	<input type="checkbox"/>	ANT
 	alm.phase.extractBundle	true		<input checked="" type="checkbox"/>	<input type="checkbox"/>	None

[Create Parameter](#)

- Click the  *Edit* icon in front of the parameter whose value you want to modify. The *Edit Phase Parameter* window will be displayed.

Edit Phase Parameter

Phase com.ikanalm.phases.database.general.updateDB-1.0.0

Secure No

Name rdbms.type

Integration Type None

Default value ORACLE

Description The database type. One of the following: HSQLDB, MYSQL, MSSQL, ORACLE or DB2

Mandatory Yes No

[Save](#) [Reset](#) [Cancel](#)

- Modify the *Default value* and click *Save*.
The modified default value will be displayed in the Phase Parameters overview.
When linking the Phase to an Environment, all default values will be set as Environment Phase Parameter values.
The Environment Phase Parameter value is the parameter value that will be used when executing a Level Request using the Phase.

Environment Phase Parameters










5.1. The Environment Phases Overview


When a Phase is linked to a Build and/or Deploy Environment, the initial values specified in the Global Definition of the Phase ([Initial Parameter Values](#) (page 10)) are set as the values of the Environment Phase Parameters.

You can verify the value of the Environment Phase Parameters on the *Phases Parameters Overview* in the Build or Deploy Environment section for the Project.


1. On the main menu, select *Project Administration*.
2. On the Projects Overview panel, click the  *Edit* icon in front of the Project.
3. Select *Build (Deploy) Environments > Overview* from the submenu.

The *Build (Deploy) Environment Phases Overview* will be displayed.

Deploy Environments Overview												
			Name	Machine	Source Location	Target Location	Deploy Script	Partial Deploy	Level	Deploy Tool	Build Environment	Debug
			proddeploy	docalm	D:/ALMDemo_57/environments/Customers/PRODUCTION/proddeploy/source	D:/ALMDemo_57/environments/Customers/PRODUCTION/proddeploy/target	deployProdLevel.xml		PRODUCTION	ANT1.9.3	contbuild	
			testdeploy	ikandemo004v	D:/ALMDemo_57/environments/DemoProject/TEST/testdeploy/source	D:/ALMDemo_57/environments/DemoProject/TEST/testdeploy/target	deployTestLevel.xml		TEST	ANT1.9.3	contbuild	
			uatdeploy	ikandemo004v	D:/ALMDemo_57/environments/Customers/UAT/uatdeploy/source	D:/ALMDemo_57/environments/Customers/UAT/uatdeploy/target	deployTestLevel.xml		UAT	ANT1.9.3	contbuild	

4. On the *Build (Deploy) Environments Overview* panel, click the  *Edit Phases* icon in front of the required Phase.




































The *Phases Overview* will be displayed.

Project Administration > Deploy Environment Phases Overview 


Deploy Environment


Name testdeploy	Debug No
Source Location D:/ALMDemo_57/environments/DemoProject/TEST/testdeploy/source	Level TEST
Target Location D:/ALMDemo_57/environments/DemoProject/TEST/testdeploy/target	Machine ikandemo004v
Deploy Script deployTestLevel.xml	Deploy Tool ANT 1.9.3
Partial Deploy No	Build Environment contbuild

Phases Overview

				Phase Name	Phase Version	Fail On Error	Next Phase On Error
				 Transport Build Result	5.7.0	Yes	Cleanup Build Result
				 Decompress Build Result	5.7.0	Yes	Cleanup Build Result
				 Verify Deploy Script	5.7.0	Yes	Cleanup Build Result
				 Execute Script	5.7.0	Yes	Cleanup Build Result
				 Update Database	1.0.0	Yes	Cleanup Build Result
				 Tomcat Deploy	1.0.0	Yes	Cleanup Build Result
				 Cleanup Build Result	5.7.0	No	

[Insert Phase](#) [History](#)

5. Click the  *View Parameters* icon in front of the required Phase.
The *Phase Parameter Overview* will be displayed.

Project Administration > Phase Parameter Overview 

Deploy Environment












Name	testdeploy	Debug	No
Source Location	D:/ALMDemo_57/environments/DemoProject/TEST/testdeploy/source	Level	TEST
Target Location	D:/ALMDemo_57/environments/DemoProject/TEST/testdeploy/target	Machine	ikandemo004v
Deploy Script	deployTestLevel.xml	Deploy Tool	ANT1.9.3
Partial Deploy	No	Build Environment	contbuild

Environment Phase

Phase Name	Update Database	Next Phase on Success	Tomcat Deploy
Phase Version	1.0.0	Next Phase on Error	Cleanup Build Result
Fail on Error	Yes	Label	

[Phases Overview](#)

Phase Parameters

	Name	Value	Integration Type	Mandatory	Secure
	alm.phase.builder		ANT	<input checked="" type="checkbox"/>	
	alm.phase.extractBundle	true	None	<input checked="" type="checkbox"/>	
	alm.phase.mainScript	UpdateDB.xml	None	<input checked="" type="checkbox"/>	
	rdbms.dbname	almDemo	None	<input checked="" type="checkbox"/>	
	rdbms.dbschema		None		
	rdbms.port		None		
	rdbms.pwd	*****	None	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
	rdbms.server	localhost	None	<input checked="" type="checkbox"/>	
	rdbms.type	MYSQL	None	<input checked="" type="checkbox"/>	
	rdbms.user	sa	None	<input checked="" type="checkbox"/>	
	sql.script	\${source}/sql/generateTestDatabase.sql	None	<input checked="" type="checkbox"/>	

11 items found, displaying all

5.2. Modifying the Environment Phase Parameter Value

The recommended way of working is to adapt the initial values imported BEFORE you link the Phase to an Environment.

After having linked the Phase to one or more Environments, it is still possible to modify the value on Project Level.

1. Go to the *Phase Parameter Overview* screen for the required Phase, as described in the previous section ([The Environment Phases Overview](#) (page 12).)

Project Administration > Phase Parameter Overview ?

Deploy Environment			
Name	testdeploy	Debug	No
Source Location	D:/ALMDemo_57/environments/DemoProject/TEST/testdeploy/source	Level	TEST
Target Location	D:/ALMDemo_57/environments/DemoProject/TEST/testdeploy/target	Machine	ikandemo004v
Deploy Script	deployTestLevel.xml	Deploy Tool	ANT1.9.3
Partial Deploy	No	Build Environment	contbuild

Environment Phase			
Phase Name	Update Database	Next Phase on Success	Tomcat Deploy
Phase Version	1.0.0	Next Phase on Error	Cleanup Build Result
Fail on Error	Yes	Label	

[Phases Overview](#)

Phase Parameters						
	Name	Value	Integration Type	Mandatory	Secure	
	alm.phase.builder		ANT	<input checked="" type="checkbox"/>		
	alm.phase.extractBundle	true	None	<input checked="" type="checkbox"/>		
	alm.phase.mainScript	UpdateDB.xml	None	<input checked="" type="checkbox"/>		
	rdbms.dbname	almDemo	None	<input checked="" type="checkbox"/>		
	rdbms.dbschema		None			
	rdbms.port		None			
	rdbms.pwd	*****	None	<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>
	rdbms.server	localhost	None	<input checked="" type="checkbox"/>		
	rdbms.type	MYSQL	None	<input checked="" type="checkbox"/>		
	rdbms.user	sa	None	<input checked="" type="checkbox"/>		
	sql.script	\${source}/sql/generateTestDatabase.sql	None	<input checked="" type="checkbox"/>		

11 items found, displaying all

2. Click the *Edit* icon in front of the parameter.
The *Edit Parameter Value* window will be displayed.

Edit Parameter Value
✖

Name rdbms.type

Value

Save
Reset
Cancel


3. Modify the value of the parameter and click *Save*.
The value of the Environment Phase Parameter will now be modified for that particular environment only.

4. Alternative method for modifying the parameter value for more than one environment:

On the *Phase Parameters* panel, click the *Edit Global Phase Parameter* icon in front of the parameter.

Clicking this icon will display the *Edit Phase Parameter* window showing all Environments the Phase Parameter is used on.

Note: For this functionality, you need Global Administrator Access Rights.

5. Next, you can use the  *Edit Environment Phase Parameter* links on the *Connected Environment Parameters* panel as a shortcut to easily switch to the same parameter on another Environment. This is particularly easy if you need to modify several parameters.



Project	Level	Environment	Parameter Value
DemoProject	CONTBUILD	contbuild	MYSQL
DemoProject	PRODUCTION	proddeploy	ORACLE
DemoProject	TEST	testdeploy	MYSQL
DemoProject	UAT	uatdeploy	\${ikandemo004v.rdbms.type}

Note: If you have Global Administrator Access Rights, you can modify the value of Environment Phase Parameters for several environments at once using the *Mass Edit* functionality in the Global Definition of the Phase. For more information, refer to the *IKAN ALM User Guide*.


Machine Parameters

If a parameter value is valid for an entire machine, it might be interesting to create a Machine Parameter.

Note: When used in combination with Ant Phases, the name of the Machine Parameter must be unique as Ant will not accept a reference to a parameter with the same name.

6.1. Creating a Machine Parameter

Note: For creating Machine Parameters, you need Global Administrator Access Rights.

1. Select *Global Administration > Machines > Machine Parameters*.
2. Click the  *Create Parameter* link in front of the Machine to display the *Create Machine Parameter* window.



3. Fill out the fields for the new Machine Parameter.
The following fields are available. The *Key* field is mandatory:

Field	Meaning
Machine	This field displays the current Machine.
Secure	This field indicates whether the Parameter is secured or not.

Field	Meaning
Key	<p>In this field, enter the Key (Name) for the Machine Parameter.</p> <p>Important: this name must be unique.</p> <p>We recommend you to use the name of the original Phase parameter preceded by the machine name. That way it will be easy to track where the value used by the Level Request has been taken from, e.g. <code>ikana1m004v.rdbms.type</code>.</p>
Value	<p>In this field, enter the value(s) for the new Machine Parameter.</p> <p>The following possibilities apply:</p> <ul style="list-style-type: none"> • Enter the fixed value, if you are creating a non-editable Machine Parameter. • Enter the default value, if you are creating an editable Machine Parameter. • Enter the list of possible values, separated by a semicolon (;), if you are creating a dynamic Machine Parameter (for example: <code>DB2;ORACLE;HSQLB</code>). Subsequently, these values can be selected from a drop-down list when creating a Level Request
Repeat Value	Required field for secured Machine Parameters: repeat the secured value.
Description	<p>In this field, enter a description for the Parameter.</p> <p>Note: We recommend you to put in the description which Phase is using this parameter.</p>
Mandatory	If the original Phase Parameter is mandatory, this new Machine Parameter should be set to mandatory as well.
Editable	<ul style="list-style-type: none"> • Select <i>Yes</i>, if the new Machine Parameter must be defined as editable. When you create a Level Request, you can accept the default value (the one you enter in the Value field during creation) or specify the value of your choice for the Parameter. • Select <i>No</i>, if the new Machine Parameter should not be defined as editable. When you create a Level Request, only the preset value (the one you entered in the Value field during creation) for this Parameter can be offered to the Build/Deploy Script. <p>This field is not provided for secured Machine Parameters.</p>
Dynamic	<ul style="list-style-type: none"> • Select <i>Yes</i>, if the new Machine Parameter must be defined as dynamic. When you create a Level Request, you can select one of the predefined values from the drop-down list. These are the values you enter in the Value field during creation and which you separate by a semicolon (;). The selected value will be offered to the Build/Deploy Script. The first value in this list will be the default value, i.e., the first value on the drop-down list. • Select <i>No</i>, if the new Machine Parameter should not be defined as dynamic. <p>This field is not provided for secured Machine Parameters.</p>

- Click *Create* to confirm the creation of the Machine Parameter.

The screenshot shows a 'Create Machine Parameter' dialog box with the following details:


- Machine:** ikandemo004v
- Secure:** Yes No
- Key:** ikandemo004v.rdbms.type *
- Value:** HSQLDB
- Description:** Used by the UpdateDB Phase
- Mandatory:** Yes No
- Editable:** Yes No
- Dynamic:** Yes No
- Buttons:** Create, Reset, Cancel

6.2. Replacing the Value of the Environment Phase Parameter

As explained before, the value of the Environment Phase Parameter is always used when executing the Phase. To enforce the usage of the Machine Parameter, you have to create a reference to its value.

- Go to the *Phase Parameter Overview* as explained in the section [Modifying the Environment Phase Parameter Value](#) (page 14)

- Click the  *Edit* icon in front of the parameter.

Project Administration > Phase Parameter Overview 

Deploy Environment
















Name	testdeploy	Debug	No
Source Location	D:/ALMDemo_57/environments/DemoProject/TEST/testdeploy/source	Level	TEST
Target Location	D:/ALMDemo_57/environments/DemoProject/TEST/testdeploy/target	Machine	ikandemo004v
Deploy Script	deployTestLevel.xml	Deploy Tool	ANT1.9.3
Partial Deploy	No	Build Environment	contbuild

Environment Phase

Phase Name	Update Database	Next Phase on Success	Tomcat Deploy
Phase Version	1.0.0	Next Phase on Error	Cleanup Build Result
Fail on Error	Yes	Label	

[Phases Overview](#)

Phase Parameters

	 Name	Value	 Integration Type	 Mandatory	 Secure
	alm.phase.builder		ANT	<input checked="" type="checkbox"/>	
	alm.phase.extractBundle	true	None	<input checked="" type="checkbox"/>	
	alm.phase.mainScript	UpdateDB.xml	None	<input checked="" type="checkbox"/>	
	rdbms.dbname	almDemo	None	<input checked="" type="checkbox"/>	
	rdbms.dbschema		None		
	rdbms.port		None		
	rdbms.pwd	*****	None	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
	rdbms.server	localhost	None	<input checked="" type="checkbox"/>	
	rdbms.type	MYSQL	None	<input checked="" type="checkbox"/>	
	rdbms.user	sa	None	<input checked="" type="checkbox"/>	
	sql.script	\${source}/sql/generateTestDatabase.sql	None	<input checked="" type="checkbox"/>	

11 items found, displaying all

The *Edit Parameter Value* window will be displayed.

Edit Parameter Value ✖

Name	rdbms.type
Value	<input style="width: 80%;" type="text" value="MYSQL"/>
<input type="button" value="Save"/> <input type="button" value="Reset"/> <input type="button" value="Cancel"/>	

- Replace the value of the parameter with a reference to the Machine Parameter and click Save. The syntax of the parameter is: `${replacingParameter}`. In our example, the reference would be `${ikana1mdemo004v.rdbms.type}`.

Edit Parameter Value ✖

Name	rdbms.type
Value	<input style="width: 80%;" type="text" value="\${ikana1mdemo004v.rdbms.type}"/>
<input type="button" value="Save"/> <input type="button" value="Reset"/> <input type="button" value="Cancel"/>	

When creating the Level Request, the value of the Machine Parameter will replace the value of the Environment Phase Parameter.

Note: For more information on alternative ways of editing the value of the Environment Phase Parameter, refer to the section [Modifying the Environment Phase Parameter Value](#) (page 14).

Environment Parameters


Parameters can be further refined on Environment level.

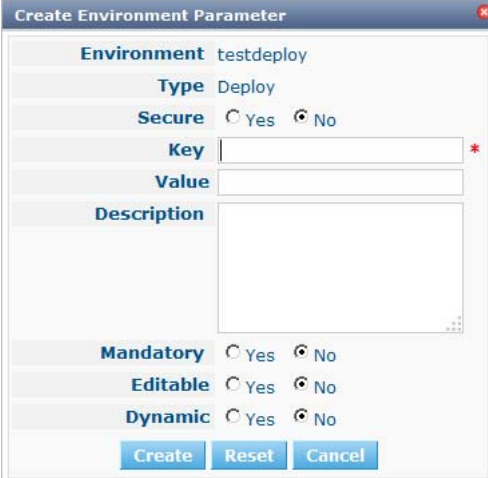
Here, the same principle applies as for the Machine Parameters.

Note: When used in combination with Ant Phases, the name of the Environment Parameter must be unique as Ant will not accept a reference to a parameter with the same name.

7.1. Creating an Environment Parameter

Note: For creating Environment Parameters, you need Global Administrator Access Rights.

1. Go to the *Projects Overview* in the *Project Administration* section and select the required project.
2. From the submenu, select *Build (Deploy) Environments* > *Build (Deploy) Parameters*.
3. Click the  *Create Parameter* link in front of the Environment for which you want to create a Parameter.



4. Fill out the fields for the Environment Parameter.
The following fields are available. The *Key* field is mandatory:

Field	Meaning
Environment	This field displays the name of the current Environment.
Type	This field displays the type of Parameter being created: <i>Build</i> or <i>Deploy</i> .
Secure	This field indicates whether the Parameter is secured or not.
Key	In this field, enter the Key (Name) for the Environment Parameter. Important: this name must be unique. We recommend you to use the name of the original Phase parameter preceded by the environment name. That way it will be easier to track where the value used by the Level Request has been taken from, e.g. <code>testdeploy.rdbms.type</code> .
Value	In this field, enter the value(s) for the new Environment Parameter. The following possibilities apply: <ul style="list-style-type: none"> • Enter the fixed value, if you are creating a non-editable Environment Parameter. • Enter the default value, if you are creating an editable Environment Parameter. • Enter the list of possible values, separated by a semicolon (;), if you are creating a dynamic Environment Parameter (for example: <code>DB2;ORACLE;HSQLB</code>). Subsequently, these values can be selected from a drop-down list when creating a Level Request
Repeat Value	Required field for secured Environment Parameters: repeat the secured value.
Description	In this field, enter a description for the Parameter. Note: We recommend you to put in the description which Phase is using this parameter.
Mandatory	If the original Phase Parameter is mandatory, this new Environment Parameter should be set to mandatory as well.
Editable	<ul style="list-style-type: none"> • Select <i>Yes</i>, if the new Environment Parameter must be defined as editable. When you create a Level Request for this Environment, you can accept the default value (the one you enter in the Value field during creation) or define a value yourself for this Environment Parameter. • Select <i>No</i>, if the new Environment Parameter should not be defined as editable. When you create a Level Request for this Environment, only the preset value (the one you enter in the Value field during creation) for this Environment Parameter can be offered to the Script. This field is not provided for secured Environment Parameters.
Dynamic	<ul style="list-style-type: none"> • Select <i>Yes</i>, if the new Environment Parameter must be defined as dynamic. When you create a Level Request for this Environment, you can select one of the predefined values from the drop-down list. These are the values you enter in the Value field during creation and which you separate by a semicolon (;). The selected value will be offered to the Script. The first value in this list will be the default value, i.e., the first value on the drop-down list. • Select <i>No</i>, if the new Environment Parameter should not be defined as dynamic. This field is not provided for secured Environment Parameters.

- Click *Create* to confirm the creation of the Environment Parameter.

The screenshot shows a 'Create Environment Parameter' dialog box with the following fields and values:

- Environment:** testdeploy
- Type:** Deploy
- Secure:** Yes No
- Key:** testdeploy.rdbms.type *
- Value:** DB2;ORACLE;HSQLDB
- Description:** Used by the UpdateDB Phase
- Mandatory:** Yes No
- Editable:** Yes No
- Dynamic:** Yes No


Buttons at the bottom: Create, Reset, Cancel.

7.2. Replacing the Value of the Environment Phase Parameter

Just as for the Machine Parameters, the value of the Environment Parameter will be overridden by the value of the Environment Phase Parameter is when executing the Phase. To enforce the usage of the Environment Parameter, you have to create a reference to its value.

- Go to the *Phase Parameter Overview* as explained in the section [Modifying the Environment Phase Parameter Value](#) (page 14)

- Click the  *Edit* icon in front of the parameter.

Project Administration > Phase Parameter Overview 

Deploy Environment












Name	testdeploy	Debug	No
Source Location	D:/ALMDemo_57/environments/DemoProject/TEST/testdeploy/source	Level	TEST
Target Location	D:/ALMDemo_57/environments/DemoProject/TEST/testdeploy/target	Machine	ikandemo004v
Deploy Script	deployTestLevel.xml	Deploy Tool	ANT1.9.3
Partial Deploy	No	Build Environment	contbuild

Environment Phase

Phase Name	Update Database	Next Phase on Success	Tomcat Deploy
Phase Version	1.0.0	Next Phase on Error	Cleanup Build Result
Fail on Error	Yes	Label	

[Phases Overview](#)

Phase Parameters

	Name	Value	Integration Type	Mandatory	Secure
	alm.phase.builder		ANT	<input checked="" type="checkbox"/>	
	alm.phase.extractBundle	true	None	<input checked="" type="checkbox"/>	
	alm.phase.mainScript	UpdateDB.xml	None	<input checked="" type="checkbox"/>	
	rdbms.dbname	almDemo	None	<input checked="" type="checkbox"/>	
	rdbms.dbschema		None		
	rdbms.port		None		
	rdbms.pwd	*****	None	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
	rdbms.server	localhost	None	<input checked="" type="checkbox"/>	
	rdbms.type	MYSQL	None	<input checked="" type="checkbox"/>	
	rdbms.user	sa	None	<input checked="" type="checkbox"/>	
	sql.script	\${source}/sql/generateTestDatabase.sql	None	<input checked="" type="checkbox"/>	

11 items found, displaying all

The *Edit Parameter Value* window will be displayed.

Edit Parameter Value ✖

Name rdbms.type

Value

- Replace the value of the parameter with a reference to the Environment Parameter and click Save.

The syntax of the parameter is: `${replacingParameter}`.

In our example, the reference would be `${testdeploy.rdbms.type}`.

Edit Parameter Value ✖

Name rdbms.type

Value



When creating the Level Request, the value of the Machine Parameter will replace the value of the Environment Phase Parameter.

Note: For more information on alternative ways of editing the value of the Environment Phase Parameter, refer to the section [Modifying the Environment Phase Parameter Value](#) (page 14).

Overview Connected Environment Parameters

Note: To use this functionality, you need Global Administrator Access Rights.

An overview of all values defined on the environments connected to a Phase, can be found on the *Mass Edit Phase Parameter* window which can be accessed via the Global Definition of the Phase in the Global Administration section.

1. Go to the *Global Administration > Phases > Overview*.
2. Click the  *Edit* icon in front of the required Phase.
3. Next, click the  *Mass Edit* icon in front of the required parameter.

The *Mass Edit Phase Parameter* window is displayed. Here you can see for each of the Environments which parameter will be used when creating the Level Request.

Phase com.ikanalm.phases.database.general.updateDB-1.0.0 **Secure** No

Name rdbms.type **Integration Type** None

Default value MYSQL **Mandatory** Yes

Description The database type. One of the following: HSQLDB, MYSQL, MSSQL, ORACLE or DB2

Connected Environment Parameters

<input type="checkbox"/>	Project	Level	Environment	Parameter Value
<input type="checkbox"/>	DemoProject	CONTBUILD	contbuild	\${ikandemo004v.rdbms.type}
<input type="checkbox"/>	DemoProject	PRODUCTION	proddeploy	MYSQL
<input type="checkbox"/>	DemoProject	TEST	testdeploy	\${testdeploy.rdbms.type}
<input type="checkbox"/>	DemoProject	UAT	uatdeploy	\${ikandemo004v.rdbms.type}

Set value for selection **Set**

Reset value for selection to default **Reset**

Cancel

To know the actual value that has been used for a specific Level Request, you can consult the Phase Logs ([Phase Logs](#) (page 29)).

Note: On this window, you can also mass-edit the parameter values for several environments at once. For more information, refer to the *IKAN ALM User Guide*.

CHAPTER 9

Phase Logs

To verify the actual value that has been used when creating the Level Request, you can display the Phase Logs on the *Level Request Detail* screen.

1. After having created a Level Request, you can click the *OID* link on the *Level Requests Overview* to display the *Level Request Detail* screen.
2. Next, select the *Phase Logs* tab and select the Build or Deploy Phase from the list.

Desktop>Level Request Detail

Success DemoProject / H_1-0 / TEST / Build# 7
74: Deliver to Test
Requested by: global on: 17-10-16 14:08:59

Summary **Phase Logs** Results Approvals Issues Sources Modifications Dependencies

[Back](#) | [Refresh](#) | [Build History](#)

Phase Logs

> Level Parameters

Phase Name	Start Date/Time	Duration
> + Retrieve Code	17-10-16 14:08:59	< 1 sec.
> + Build	17-10-16 14:08:59	< 1 sec.
> + Tag Code	17-10-16 14:08:59	< 1 sec.
> + Deploy	17-10-16 14:08:59	00:00:06
▼ + Deploy 54 on machine ikandemo004v	17-10-16 14:08:59	00:00:06

OID 54 Start Date/Time 17-10-16 14:08:59
Environment testdeploy Duration 00:00:06
Machine ikandemo004v Status Success

> Deploy Parameters

> + Transport Build Result	17-10-16 14:09:00	< 1 sec.
> + Decompress Build Result	17-10-16 14:09:00	< 1 sec.
> + Verify Deploy Script	17-10-16 14:09:00	< 1 sec.
> + Execute Script	17-10-16 14:09:00	00:00:01
> + Update Database	17-10-16 14:09:01	< 1 sec.
> + Tomcat Deploy	17-10-16 14:09:02	00:00:03
> + Cleanup Build Result	17-10-16 14:09:06	< 1 sec.
> + Cleanup Work Copy	17-10-16 14:09:06	< 1 sec.

3. Next, click the Phase link and display the Phase Parameters.

The screenshot shows the 'Update Database' phase expanded. A red box highlights the 'Update Database' phase header, and another red box highlights the 'Phase Parameters' sub-header. A red arrow points from the 'Update Database' header to the 'Phase Parameters' sub-header. The parameters table is as follows:

Key	Value
alm.phase.extractBundle	true
alm.phase.mainScript	UpdateDB.xml
rdbms.dbname	almDemo
rdbms.pwd	****
rdbms.server	localhost
rdbms.type	\${testdeploy.rdbms.type}
rdbms.user	sa
sql.script	\${source}/sql/generateTestDatabase.sql

4. To know the exact value of the parameter, you click the *Deploy (or Build) Parameters* link for the *Deploy (or Build)* Phase.

The screenshot shows the 'Deploy Parameters' section expanded. A red box highlights the 'Deploy Parameters' sub-header, and another red box highlights the 'testdeploy.rdbms.type' parameter value. A red arrow points from the 'Deploy Parameters' sub-header to the 'testdeploy.rdbms.type' parameter. The parameters table is as follows:

Key	Value
alm.build.environmentName	contbuild
alm.build.filename	DemoProject_H_1-0_b7_CONTBUILD_win.zip
alm.build.machineName	ikandemo004v
alm.build.number	7
alm.build.oid	20
alm.build.startDateTime	17-10-2016 13:44
alm.deploy.environmentName	testdeploy
alm.deploy.machineName	ikandemo004v
...	
alm.projectStream.type	H
applicationName	customers_test
database.host	localhost
database.name	almDemo
database.pwd	****
database.user	almtest
ikandemo004v.rdbms.type	HSQLDB
source	D:/ALMDemo_57/environments/DemoProject/TEST/testdeploy/source/54
target	D:/ALMDemo_57/environments/DemoProject/TEST/testdeploy/target
testdeploy.rdbms.type	DB2