

# RTC-UCD terminology

The following list of terms is an overview of terms in select areas of IBM Rational Team Concert (RTC) and IBM UrbanCode Deploy (UCD).

## Contents

Automation (UCD).....	2
SCM (RTC) .....	3
SCM (UCD).....	4
Build (RTC).....	5
Security (RTC).....	6
Security (UCD) .....	6
Notices.....	7



## Automation (UCD)

### Application

UCD applications are collections of UCD components that work together to provide a business function. Applications include application processes, which deploy the components. Applications also contain environments, which represent the target systems that the application components are deployed on.

UCD components are assigned to applications. However, components are separate from applications; each component can be used in any number of applications.

IBM KC (UCD): [Applications](#)

### Compliance

Compliance refers to the number of UCD component versions that were successfully deployed to an environment. For example, if an application process attempted to deploy 3 component versions to an environment and only 2 deployed successfully, the compliance for the environment is 2/3.

IBM KC (UCD): [Environment inventory and compliance](#)

### Deployment

In UCD, the general term “deployment” usually refers to deploying an application by running an application process.

IBM KC (UCD): [Deploying](#)

### Environment

UCD environments are groups of UCD resources (mainly, target systems and the agents on those systems) that host application components. Environments are contained by applications.

IBM KC (UCD): [Environments](#)

### Inventory

Primarily, an inventory is a list of UCD component versions that were deployed to an environment, whether or not those component versions deployed successfully. The idea of inventory is closely related to the idea of compliance.

Each UCD component also has an inventory that lists every place that the component is deployed. Environments and resources have inventories that list every UCD component in that environment or resource.

IBM KC (UCD): [Environment inventory and compliance](#)

### Process

Component processes deploy UCD components, application processes deploy collections of UCD components, and generic processes run automated tasks on target systems.

IBM KC (UCD): [Processes](#)

### Resource

UCR resources are representations of external objects on the UCD server. For example, agents are represented by agent resources on the server.

IBM KC (UCD): [Resources](#)

### Source configuration

The source configuration is the collection of settings that describe how the UCD server imports build output into UCD component versions. Source configuration plugins provide connectivity from the UCD server to build and SCM systems such as Git and Subversion so the server can retrieve build output from those systems.

IBM KC (UCD): [Creating components](#)

## SCM (RTC)

### Accept

To load RTC change sets that other people delivered to a stream. Similar to “pull” or “update” in other source code management systems.

IBM KC (RTC): [Change sets and change flow](#)

### Baseline

A permanent copy of an RTC component in a particular workspace or stream. Baselines are a known initial state from which teams can start work.

IBM KC (RTC): [Component baselines](#)

### Change set

One or more changes to RTC component files that can be delivered to streams.

IBM KC (RTC): [Change sets and change flow](#)

### Check in

To add file changes to a change set.

IBM KC (RTC): [Change sets and change flow](#)

### Component

RTC components are collections of related artifacts such as source files that have the same life cycle. Components are added to streams to store the source files that are being worked on in that stream.

IBM KC (RTC): [Streams and components](#)

### Deliver

The process of sending change sets from a repository workspace to one or more streams. Similar to “push” or “commit” in other source control systems.

IBM KC (RTC): [Change sets and change flow](#)

### Flow target

The stream to which change sets are delivered.

IBM KC (RTC): [Flow targets](#)

### Load

To check out RTC components so you can work with them in a local workspace.

IBM KC (RTC): [Change sets and change flow](#)

### Repository workspace

In RTC source control, a repository workspace (or sometimes just “workspace”) is a user’s collection of components that the user is working with. Users deliver changes from their repository workspace to one or more streams.

IBM KC (RTC): [Workspaces and sandboxes](#)

### Sandbox

An isolated reflection of a repository workspace where a developer can modify and test items away from other people.

IBM KC (RTC): [Workspaces and sandboxes](#)

### Snapshot

RTC snapshots capture the state of a stream or repository at a point in time. Technically, RTC snapshots are a collection of all baselines in a stream or repository. RTC snapshots can be imported into UCD as component versions.

IBM KC (RTC): [Workplace snapshots](#)

### Stream

In RTC source control, streams contain components and store the changes to the component artifacts while developers work to a new version or release of a piece of work.

IBM KC (RTC): [Creating streams](#)

### Version

In RTC, the term “version” is usually used in a more general sense to refer to a specific state of an artifact. The version of source artifacts in RTC are not directly connected to UCD component versions unless the RTC-UCD integration uses the RTC source version name as the UCD component version name.

IBM KC (RTC): [Workplace snapshots](#)

### Work item

Work items represent tasks, plan items, or defects to assign to people to complete or resolve.

IBM KC (RTC): [Work items](#)

## SCM (UCD)

### **CodeStation**

The UCD artifact repository is called CodeStation, and it stores UCD component version artifacts and provides them at deployment time.

IBM KC (UCD): [Component versions](#)

### **Component**

UCD components represent deployable artifacts, usually not source code. UCD components include component processes, which automate deployment of the artifacts, and component versions, which are discrete collections of artifacts that represent a version of that component.

IBM KC (UCD): [Components](#)

### **Snapshot**

A UCD snapshot is a collection of UCD component versions that are known to work together. Snapshots are useful for storing working deployments and for promoting working deployments out of a development environment into a testing or production environment.

IBM KC (UCD): [Snapshots](#)

### **Version**

In UCD, the term “version” usually refers to component versions, which are discrete instances of the artifacts in a component. Component versions are usually imported from build systems such as RTC. Components can have full versions, which provide a complete copy of all artifacts, or incremental versions, which provide only the changed files from the prior version.

IBM KC (UCD): [Component versions](#)

### **Work item**

UCD refers to pending approvals as “work items.”

IBM KC (UCD): [Work items](#)

## Build (RTC)

### **Agent (Jazz Build Engine, Build Forge Agent)**

Agents run on target systems and allow the RTC server to automate tasks, such as build processes. For RTC, these agents include the Jazz Build Engine and the Build Forge agent.

IBM KC (RTC): [Rational Build Agent](#)

### **Build engine**

The build engine is a virtual representation of a build agent on the RTC server. One build engine on RTC can map to one or more instances of the Build System Toolkit.

IBM KC (RTC): [Jazz Build Engines](#)

### **Build System Toolkit**

A set of command-line tools that processes build requests from RTC. It includes the Jazz Build Engine, a collection of Ant tasks, and the build agent that handles builds on z/OS and IBM i. Each instance of the Build System Toolkit is connected to a build engine on an RTC server.

IBM KC (RTC): [Rational Team Concert Build](#)

### **Engine**

In RTC, “engine” can refer to the Jazz Build Engine or to build engines.

IBM KC (RTC): [Jazz Build Engines](#)

### **Jazz Build Engine**

The Jazz Build Engine is a command-line tool that processes build requests on distributed platforms.

IBM KC (RTC): [Jazz Build Engines](#)

### **Package**

A group of deployable artifacts.

IBM KC (RTC): [Packaging and deploying with Enterprise Extensions](#)

## Security (RTC)

### Permissions

In RTC, user permissions are mostly related to the team areas that the user is a member of. Users have a role on the team, and the role controls what permissions the user has on the team area's assets.

IBM KC (RTC): [Understanding licenses, permissions, and access control](#)

### Project

Eclipse projects are containers for source artifacts on the local user's Eclipse workspace. In many cases, RTC components include one or more Eclipse projects. Not directly related to RTC project areas.

IBM KC (RTC): [Sharing Eclipse projects with your team](#)

### Project area

A project area is a major organizational area on an RTC server. It contains things like streams and team areas.

IBM KC (RTC): [Creating and modifying project areas](#)

### Role

Each RTC user has a role on each project area or team area, and that role provides access to that area. RTC security roles do not currently connect to UCD security roles.

IBM KC (RTC): [Understanding licenses, permissions, and access control](#)

### Team/team area

An RTC team area is a smaller organizational area within a project area. It contains users that are working together on tasks and the roles that they have for that team's artifacts.

IBM KC (RTC): [Creating a team area](#)

## Security (UCD)

### Permissions

Elements such as UCD components and applications are assigned to teams, and users have access to the elements that are in the same team.

IBM KC (UCD): [Managing security](#)

### Role

UCD users have a role on a team. The role specifies what type of access the user has on the team's components, applications, and other resources. Some roles provide read-only access and others provide read/write access.

IBM KC (UCD): [Roles and permissions](#)

### Team

UCD teams are a collection of users and the components, applications, and environments that those users can access.

IBM KC (UCD): [Teams](#)

## Notices

© Copyright International Business Machines Corporation  
2016-2017.

US Government Users Restricted Rights - Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

This information was developed for products and services offered in the US. This material might be available from IBM in other languages. However, you may be required to own a copy of the product or product version in that language in order to access it.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing  
IBM Corporation  
North Castle Drive, MD-NC119  
Armonk, NY 10504-1785  
US

For license inquiries regarding double-byte character set (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing  
Legal and Intellectual Property Law  
IBM Japan Ltd.  
19-21, Nihonbashi-Hakozakicho, Chuo-ku  
Tokyo 103-8510, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some jurisdictions do not allow disclaimer of express or implied warranties in certain transactions; therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Director of Licensing  
IBM Corporation  
North Castle Drive, MD-NC119  
Armonk, NY 10504-1785  
US

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

The performance data and examples cited are presented for illustrative purposes only. Actual performance results may vary depending on specific configurations and operating conditions.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

Statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

*Trademarks*

IBM, the IBM logo, and [ibm.com](http://ibm.com) are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the web at "Copyright and trademark information" at [www.ibm.com/legal/copytrade.shtml](http://www.ibm.com/legal/copytrade.shtml).

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

