Working with SDC

Contributioning to SDC

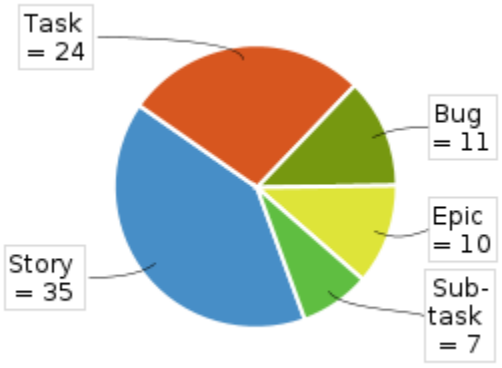
Prerequisites

Make sure to get familiar with: Developing ONAP. From future SDC dev point of view, most interesting parts are:

- Required Tools
- Setting Up Your Development Environment
- Development Procedures and Policies
- Configuring Gerrit

JIRA

SDC issues are tracked on ONAP JIRA: https://jira.onap.org/secure/RapidBoard.jspa?rapidView=71&projectKey=SDC



Jenkins

JIRA project CI: https://jenkins.onap.org/view/sdc/

Gerrit

Chenage proposals for SDC projects are available here: https://gerrit.onap.org/r/#/admin/projects/?filter=sdc

Coding Policy

ONAP Development Guides

Official development guides of ONAP: Development Guides

Contribution

license guideline

ONAP FUSS guide lines: Rules for implementing FOSS in a project

Clarification

any file that is being committed must have a a licensing header based on the ONAP guideline.

as SDC seed code came from AT&T

almost all the licensing headers will look in the following meaner:

License header

```
/*
 * Copyright (c) 2018 AT&T Intellectual Property.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *     http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
```

in case you are modifying an existing file you will need to add the flowing to the license header:

Modified license header

```
/*
 * Copyright (c) 2018 AT&T Intellectual Property.
 * Modifications Copyright (c) 2018 <Company copy right> Property.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *     http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
```

Automatic copy right addition:

Eclipse

Intlij

Code guide lines TBD

Role of SDC in ONAP - component use cases

TODO describe/link to sdc onbaording demo, describe design time

High level architecture

Architecture is described on another subpage: SDC Architecture

Subprojects

SDC is a huge component. In the road-map there is a plan to split SDC into smaller projects. To make it smooth in the future, we dont introduce new dependencies between groups of sub-projects. You can find a project and its group in the table below. There is also an informal agreement that committers have their responsibilities and focuses. If you make a change (of course, affecting subprojects from one group...), please make sure to add reviewers specialized in affected area - this will prevent the situation of having +2 without a merge. Reviewers and their responsibilities are listed below:

Committers And Roles

Subprojects

	Group	Component	Description
1	Catalog	asdctool	Collection of standalone utilities used in different projects: import, export, migration etc.

2		catalog-be	Business logic of the app. Compiles to WAR.
3		catalog-dao	Manages persistance layer: • Titan Graph DB • Cassandra • Elastic search // Contains dead Neo4j code. It may replace titan
4		catalog-fe	Proxy between ui and backend. Contains business logic as well.
5		catalog-model	Connection layer, serves models to other projects
6		catalog-ui	
7		common-app-api	Utils and logic shared between frontend and backend.
8		common-be	
9		docs	
10		security-utils	
11		sdc-titan-cassandra	
12		sdc-tosca	TOSCA parser based on jtosca. Complies with ONAP SDC TOSCA model(?)
13		sdc-workflow-designer	Graphic tool for service lifecycle management
14		services	
15		ui-ci	
16		sdc-os-chef	
17		test-apis-ci	
18		utils	
19	Onboarding	build-tools	
20		common	
21		dox-sequence-diagram-ui	
22		services	
23		onboarding	
24		openecomp-bdd	
25		openecomp-be	
26		openecomp-ui	
27	Other	jtosca	Generic TOSCA parser based on 1.1 spec
28		onap-ui	
29		dcae-d	
30		sdc-distribution-client	
31		sdc-docker-base	Docker creation project

Dependencies

SDC is a standalone application that, as such, does not require any other components to start. Deployed SDC enables to import or create an asset (design time). In order to take advantage of it (deploy a service) other components are required. For a start, please see SDC Dependencies

Interfaces - API and clients

Distribution Client

Docs available in Architecture Integration Document: SDC Distribution client AID

APIs

APIs are available on the page generated from SDC repo: http://onap.readthedocs.io/en/latest/submodules/sdc.git/docs/offeredapis.html

Documentation available in another pages: SDC API

Documentation

Documentation generated from code repository: http://onap.readthedocs.io/en/latest/submodules/sdc.git/docs/index.html

User interface

Onap home page for design related issues: Design

Tutorials

- Building and running SDC using Docker for OSX
- Categories to use in SDC project
- Consumer creation
- Csar Structure
- Deploying SDC on a Linux VM for Development
- SDC Artifact Versions
- SDC - Blessing Flow
- SDC Distribution model
- SDC docker_run Script Usage
- SDC Importing New Normatives - UNDER Construction!!!!
- SDC Jenkins Triggers
- SDC on OOM
- SDC Sanity
- SDC Simulator
- SDC supported artifact types
- SDC SWAGGER
- SDC Troubleshooting
- SDC Vagrant Common Commands
- Using Lab POD-ONAP-01 Environment
- Using Vagrant-Onap for local deployment of SDC project
- VOLTE test case input collection for tracking

Deployment

// TODO Local deployment for devs

Useful links and resources

Name	URL	Comment