Design Document

Appliance Image Meta Data

iSC Team

Table of Contents

[Overview 2](#_Toc406600588)

[Background 2](#_Toc406600589)

[Requirements 2](#_Toc406600590)

[Validations 2](#_Toc406600591)

[General iSC validations 2](#_Toc406600592)

[Minimum iSC version 2](#_Toc406600593)

[Meta-Data File Version 2](#_Toc406600594)

[Appliance Model 2](#_Toc406600595)

[Appliance Manager Type 2](#_Toc406600596)

[Appliance Manager Software Version 3](#_Toc406600597)

[Virtualization Type 3](#_Toc406600598)

[Virtualization Version 3](#_Toc406600599)

[Data extracted from Meta file 3](#_Toc406600600)

[CPU information 3](#_Toc406600601)

[Memory information 3](#_Toc406600602)

[Disk information 3](#_Toc406600603)

[Image Name 3](#_Toc406600604)

[Backward Compatible Considerations 3](#_Toc406600605)

[UI Considerations 3](#_Toc406600606)

[API Considerations 3](#_Toc406600607)

[Sample Meta Data file 4](#_Toc406600608)

# Overview

## Background

This document proposes a way to describe appliance images which need to be uploaded into iSC for deployment within a virtualized data center.

Currently in iSC, the image information is entered by the user when the image file is uploaded onto iSC. This approach is error prone since iSC has no way of validating the information.

If the Model under which the image has been uploaded does not match the actual model of the image, the user is not made aware of this mismatch until after we try to deploy the image.

We have similar challenges with respect to Manager version, manager type etc.

## Requirements

We need a Meta data file which describes the appliance image files so that the image file can be validated by iSC. In case of Openstack based appliances, we need to have the virtual hardware requirements for the appliance image as well since no Meta file like an ovf exists for Openstack.

The meta file will need to be packaged as part of the device image bundle zip file. In case the meta file is missing, we can fall back on the old UI and allowing user to enter the data.

# Validations

Here are some of the validations which we can use the Meta file for

## General iSC validations

### Minimum iSC version

iSC would validate that the image CPA will be able to communicate with the current version of iSC. Basically, if the image CPA requires a newer version of iSC, we would not allow the image upload.

### Meta-Data File Version

The Meta data file version is included for extensibility. iSC would validate that the meta file version is supported by iSC.

### Appliance Model

Currently in iSC whenever an image file is uploaded it is the user responsibility to make sure the image file matches the ‘Model’, version and manager type of the appliance is entered the right way.

The Meta file would describe the model the image file can be used against. iSC would validate the image Meta data matches the Appliance Model the image is uploaded under.

### Appliance Manager Type

The Manager type the image expects in the Meta file should match the Appliance Manager type its uploaded under

### Appliance Manager Software Version

The Appliance Manager Software Version the image expects in the Meta file should be greater than or equal to the Appliance Manager Software Version it’s uploaded under

### Virtualization Type

iSC would validate that the virtualization type specified in the meta file exists in iSC and will automatically fill in the virtualization type of the image

### Virtualization Version

iSC would validate that the virtualization version specified in the meta file exists in iSC and will automatically fill in the virtualization version of the image.

# Data extracted from Meta file

### CPU information

The minimum number of CPU’s required by the appliance. We can use this information to create the corresponding flavor in openstack.

### Memory information

The minimum memory required by the appliance. We can use this information to create the corresponding flavor in openstack.

### Disk information

The minimum disk required by the appliance. We can use this information to create the corresponding flavor in openstack.

### Image Name

The image name packaged as part of the appliance software version. iSC would validate that this file exists in the appliance zip file

### Encapsulation Types

For a manager plugin type which does not support Policy Mapping, the encapsulation type field should not be present in the meta file.

# Backward Compatible Considerations

### UI Considerations

As the 1.0 release has support for manually adding a software version image, we need to gracefully handle those cases.

We will restrict future versions of ISC to not accept images other that of VMWARE type in the manually addition case.

This would allow us to assume the memory, cpu and disk information will be available for all images within iSC which need them.

We will add a new auto import button on the UI to allow us to import all the settings along with the image.

### API Considerations

If iSC introduces a breaking change which changes either the protocol(JAXB etc) or the API(renamed some fields from the POJO response/request) then we would need to validate the CPA is compatible with the latest iSC. This should be done on the iSC side. For example, if the CPA requires an iSC of build 2000 and iSC introduced a breaking change at 2500.

If the image is uploaded to an iSC of versions inbetween 2000 and 2500, it should be fine.

If the image is uploaded to an iSC of 2501, then iSC would reject the image even if the minimum iSC version was satisfied.

# Sample Meta Data file

{

"metaDataVersion": null,

"model": null,

"managerType": null,

"managerVersion": null,

"virtualizationType": null,

"virtualizationVersion": null,

"softwareVersion": null,

"minIscVersion": {

"major": null,

"minor": null,

"build": null,

"buildTime": null,

"versionStr": null

},

"minCpus": 0,

"memoryInMb": 0,

"diskSizeInGb": 0

}