

# JJ's Reference Architecture

*Author: Jan-Joost van Zon*  
*Date: December 2014 – July 2017*  
**[Under Construction]**

## Server Architecture

### Contents

Contents .....	1
Introduction .....	1
DTAP .....	3
Folders.....	3
Development Workstation .....	4
Backups .....	5

### Introduction

The server subdivision is subject to the needs of the organization, so this overview is just a suggestion. The main concerns are safeguarding and keeping things optimal. Economics might force you to look for alternatives, but from a technical point of view the full set of servers with recommended configuration is advised. Cutting corners might make your IT run less efficiently, which would translate to cost overhead too.

<TODO: Mention the split-up into a C: and D: drive.>

<TODO: Reconsider the sizes of the development workstation drives after some cleaning up and counting disk space and considering extra dev tool requirements.>

<TODO: Consider the machine configuration needs in more detail.>

Stage	Name	Remarks	Configuration Focus Points
Development	Database server	Stores a development copy of all the databases we use.	SQL Server, decent performance, particular focus on having enough RAM.
Development	App server	Where development can use a shared FTP server if needed, run long processes to alleviate the development workstations. Can also host shared web services, be it third party, be it internally	IIS, preferably many-core. SQL Server installation is advised, for delegating number crunching from the main development database server to another server. RAM is also important, since heavy number crunching

		developed ones, even though for that last thing it is usually better to run it on the development workstations.	processes may use a lot of memory.
Development	Source control server	For storing the source control database, running the source control services, running builds, unit tests and code analysis upon each check-in.	TFS. Must be decent configuration for each checking requires a heavy process to run, and the development team has to be able to work efficiently.
Development	Workstations	Each software developer's own machine.	Two 21" monitors. No laptops, those run 2x slower. Core i5 for junior and medior developers. Core i7 for senior developers and software architects, since they will more commonly work with larger solutions. At least 8 GB RAM, so you can run large-cache applications and services. SSD for of 256 GB. Split up into a C: drive for windows <TODO: specify size> and a D: drive for data, main source code, but also aother frequently accessed things. An extra 'spinning disk' drive with at least 256 GB storage for amont other things the ability to hold large database backup files.
Development	Laptop	One laptop for a whole team, just to connect to your workstation when you are in a meeting or being on the road to a customer	Relatively low specs. Core i3, a moderate amount of RAM.
Production	Database server for number cruching	For doing the heavy processing, like datawarehouse imports and processes, heavy reporting, heavy pre-calculation processes,	Many cores

		to alleviate high-traffic production servers.	
Production	App server for number crunching	Same as above.	Many cores
Production	Database server for high traffic	For all the databases involved in high-traffic, storing data of user applications and services.	
Production	App server for high traffic	For all the production web sites and services.	Note that RAM is very relevant, to meet in-memory caching needs.
Test	Database server		
Test	App server		

## DTAP

<TODO: Write something about this. Include:

- Explain DEV, TEST and PROD. And ACC. And then also the INTERNAL and EXTERNAL production environments and their benefits, mostly with regards to publishing. Use the prefix ACC, instead of ACCEPTANCE, because the prefix goes all over the place e.g. ACC\_BuildingBlocksDB instead of ACCEPTANCE\_BuildingBlocksDB. But then again I like this better: acceptance.electroluxservice.JJ-bv.nl >

## Folders

If your servers and all your development workstations have a D: drive for data, then put the folders on the D: drive, otherwise put the the folders on the C: drive. Make a folder in the root of the drive with your company name:

D:\JJ

In this folder, each environment gets a sub-folder written in all capitals.

D:\JJ\TEST

D:\JJ\PROD

D:\JJ\DEV

Even machines with only one environment on it, should get a sub folder with the environment. That makes it better visible what environment you are working on, you can easily emulate the situation on a development workstation without additional configuration and it allows you to move environments from one server to the other.

Also add the following folders:

D:\JJ\Install

D:\JJ\Backup

<TODO: Add descriptions to each folder above.>

Those are not put in the environment folder.

The environment folder can contain the following sub folders:

D:\JJ\PROD\Images  
D:\JJ\PROD\IO Files  
D:\JJ\PROD\Log  
D:\JJ\PROD\Utilities  
D:\JJ\PROD\Web

<TODO: Add descriptions to each folder above.>

The Images folder contains images e.g. uploaded from an application, not images that are content of the application, so not icons or basic content of the web site, but user-uploaded images or images uploaded by content management systems. The Images folder can contain sub-folders to keep images apart from each other, that belong to a different application or set of applications.

D:\JJ\PROD\Images\QuestionAndAnswer  
D:\JJ\PROD\Images\Synthesizer

An application image sub-folder can contain again sub-folders, for resized images with particular dimensions:

D:\JJ\PROD\Images\QuestionAndAnswer	Contains original images.
D:\JJ\PROD\Images\QuestionAndAnswer\100x100	Images resized to 100x100 pixels
D:\JJ\PROD\Images\QuestionAndAnswer\320x280	Images resized to 320x280 pixels

The resolutions above are simply examples. You can have different sizes per application.

<TODO: Describe the Utilities folder, that you use fully qualified application names and version sub folders. Same for the Web folder, and add to that that it contains both web services as well as web applications.>

## Development Workstation

Development workstations should have the same kind of folder subdivision, also put on the same drives as on the servers. You might not put web sites in these folders, but Images, IO Files and Logs should be present in the same folder structure as the servers.

Put the source code folders in the same spot as all your coworkers. Then things keep cooperating with each other. Most of the times relative paths work, but sometimes they don't so it is a good plan to all have out copies of the source code in the same location. In case of TFS it should be D:\TFS that is mapped to the outermost root of the source control system. Not to a branch, not to a Collection, but to the server name or IP address of the source control server.

<TODO: Use this phrasing? - TODO: If servers store data files on C, then development workstations had better store it on C too, because it is incredibly confusing to have it in a

different spot in development, for instance when you try to emulate a TEST or PROD environment on your development workstation and the letter C or D can be overlooked so easily in configuration files that you keep on making mistakes.>

## **Backups**

<TODO: Write something.>