Dasar Pengembangan Sistem Informasi

System and Software Deployment

Tim Pengampu Mata Kuliah

Deployment

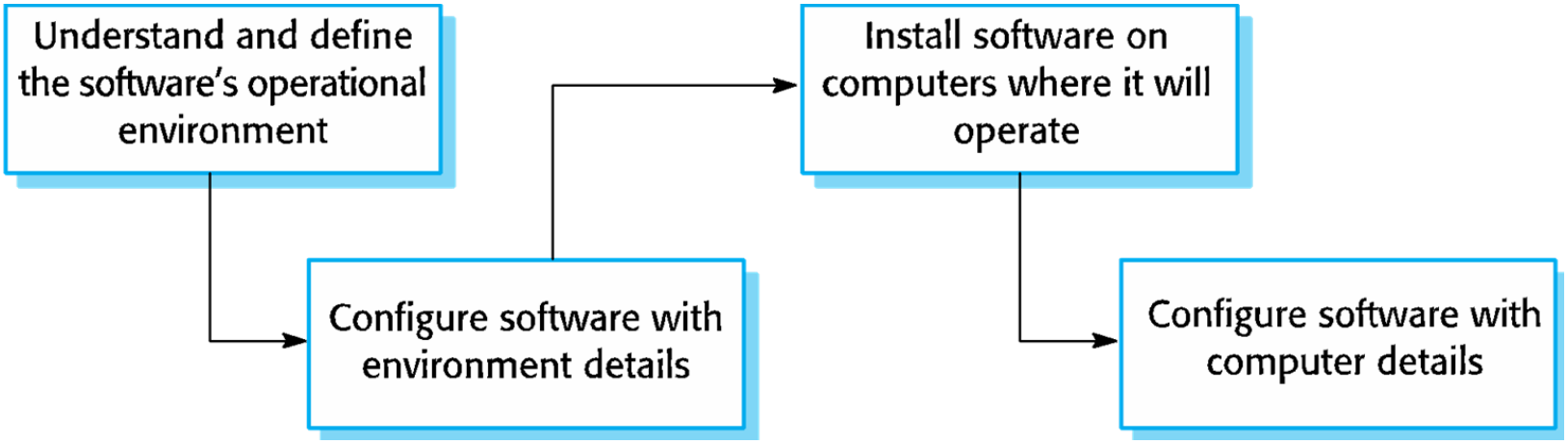
Definition

- The delivery, assembly and management at a site of the resources necessary to use a version of a software system
  - Make a software system available to its users
  - A **coherent collection of artifacts**, such as executable files, source code, data files, and documentation, that are needed at a site to offer some functionalities to the end users
- Getting software out of the hands of the developers into the hands of the users.

Definition (2)

- Software deployment is all of the activities that make a software system available for use. Examples:
  - Get the software out to the customers
  - Creating Installation Packages
  - Documentation – Installation Guide, User Manual
  - Installation
- Deployment strategies may vary depending of what kind of software we create
  - Web, Desktop, Mobile

System Deployment



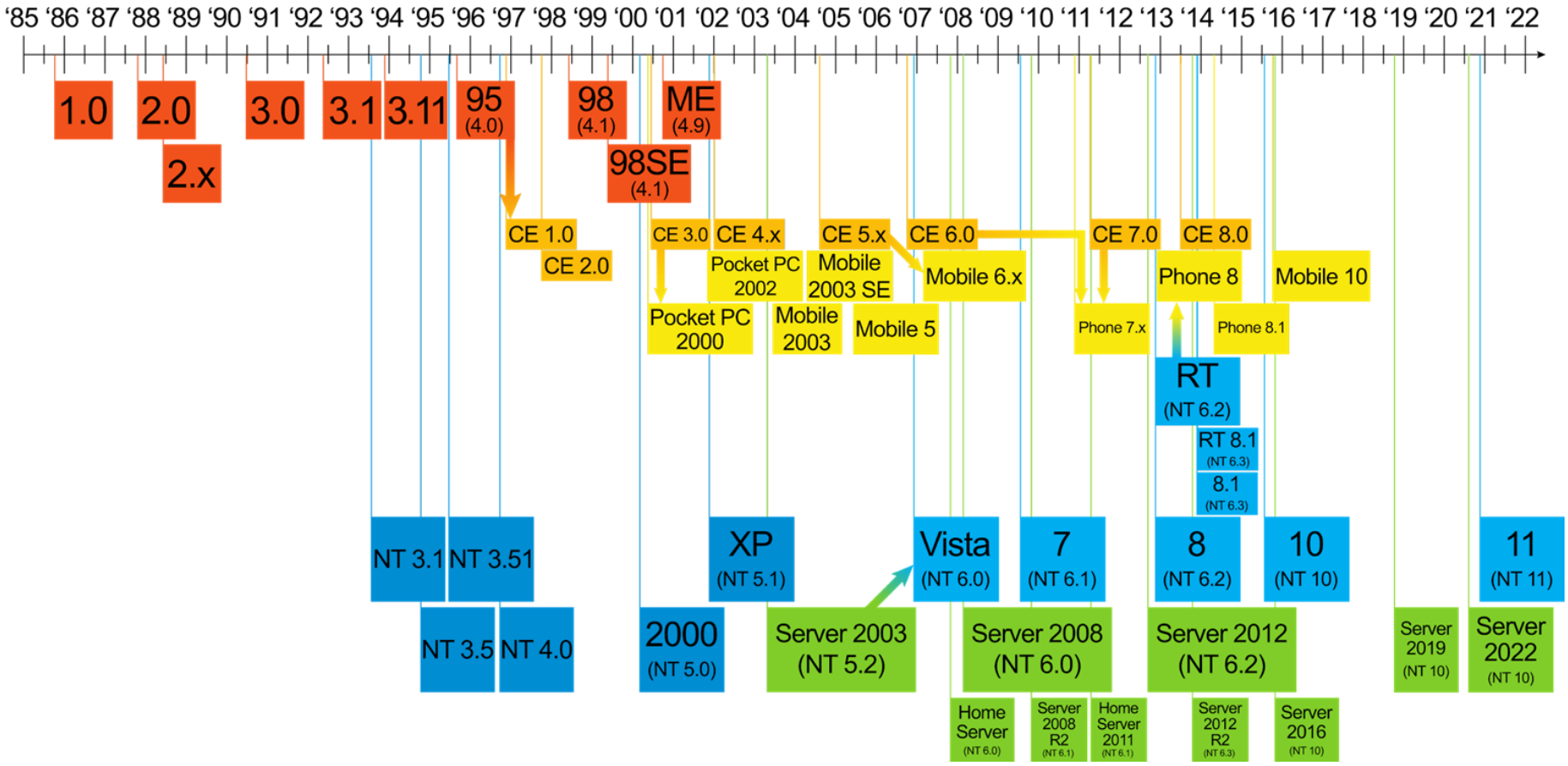
Software Releases

- **Requirements/Design/Prototype**: Plans made and approved
- **Alpha**: Foundation finished, software structure building started
- **Beta**: Main structure has been built, working on details
- **Release Candidate (RC)**: Small adjustments
- **Release To Market (RTM)**: Ready for use/sale

Maintenance Release

- After software is released
  - Patches: Small fixes
  - Service Packs: Collection of small fixes
  - Upgrades: Addon, plugins, or additional features
- Planning for next releases

Example: Windows Timeline



Why Deployment is Hard?

- More than 50% of commissioned software is not used, mostly because it fails at deployment stage.
- 80% of the cost of (commissioned) software comes at and after deployment.

Technical Deployment Issues

- Change management
  - Introducing hardware and software changes
- Dependencies management among software components
- Large-scale content distribution
  - Network reliability
- Interoperability of heterogeneous platforms

Technical Deployment Issues (2)

- Deployment coordination and customization
  - Big-bang or incremental
- Internet integration
- Security and lack of control
- Mobile app and devices heterogeneity

Deployment Issues

- **Business Processes**: Most large software systems require the customer to change the way they work.
- **Training**: No point in deploying software if the customers can't use it.
- **Support**: The need goes on, and on, and on.
- **Deployment**: How do you physically get the software installed.
- **Equipment**: Is the customer's hardware up to the job?

Deployment Issues (2)

- **Expertise**: Does the customer have the IT expertise to install the software?
- **Upgrades**: Can't avoid them!
- **Integration**: Shall the software interact/integrate with other systems of the customer.
- **Performance**: The Customer may not have the same hardware performance as in the Development/Test Environment

Development, Test, and Production Environment

"It Works on My Computer"

- Make sure to test your software on other Computers and Environments!
- Everything works on the Developer Computer, but...
  - The Customer computer is not the same as the developer's
  - The Customer may not use the same OS
  - The Customer may not use the same Web Browser

"It Works on My Computer"

- Therefore it is very important to test the software on other computers and other environments, different versions of hardware, different versions of web browsers, etc.
  - Development Environment (your computer),
  - Test Environment, and
  - Production Environment

Development, Test, Production

Typically the Developers Personal Computer with Database, Web Server and Programming Software

A clean PC/Server (or a network with PCs and Servers) where you install and test your Software. Usually set-up as Virtual Environment

The Customers environment where you install the final software (Servers and Clients)

Test Environment

- A setup of software and hardware on which the testing team is going to perform the testing.
- This setup consists of the physical setups (*hardware and software*)
  - Includes Server OS, Client OS, database server, front end running environment, browser (if web application), web server, or any other software components required to run this software product.
- The setups usually **mimics** production enviroments.

Production Environment

- The setting where software and other products are actually put into operation for their intended uses by end users.
- A real-time setting where programs are run and hardware setups are installed and relied on for organization or commercial daily operations.

Key Points

Design for Deployment

- Deployment involves configuring software to operate in its working environment, installing the system and configuring it for the operational platform.
- Vulnerabilities may be introduced at this stage as a result of configuration mistakes.
- Designing deployment support into the system can reduce the probability that vulnerabilities will be introduced.

Deployment support

- Include support for viewing and analysing configurations
- Minimise default privileges and thus limit the damage that might be caused
- Localise configuration settings
- Provide easy ways to fix security vulnerabilities
- Make sure the application is well tested
- The application should be easy to install

Questions?