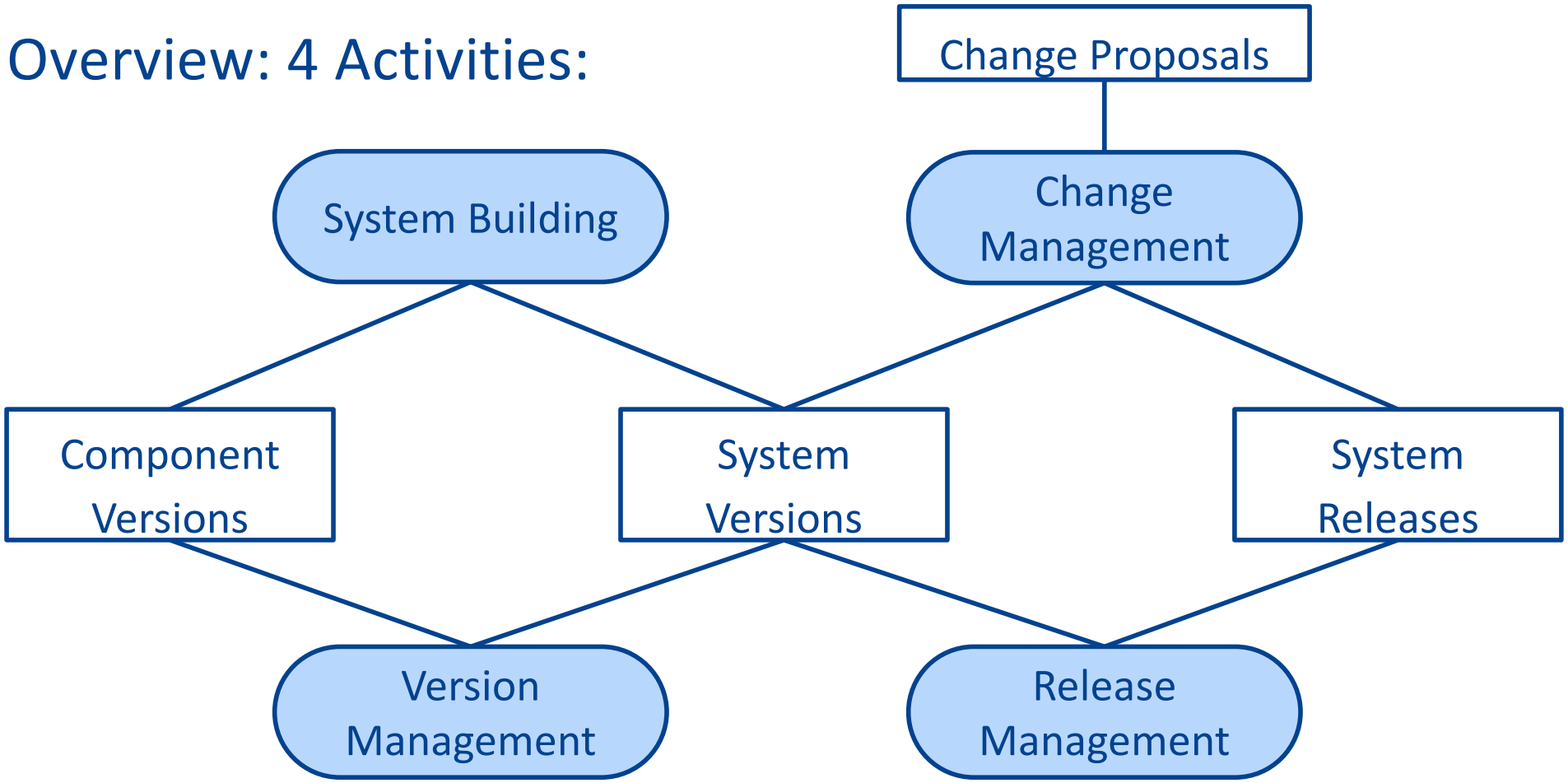
AARHUS
UNIVERSITY
DEPARTMENT OF ENGINEERING

CONFIGURATION MANAGEMENT

STEFAN HALLESTEDE
CARL SCHULTZ

CONFIGURATION MANAGEMENT

Overview: 4 Activities:



CONFIGURATION MANAGEMENT

Change management	This involves **keeping track of requests for changes** to the software from customers and developers, working out the **costs and impact of making these changes**, and deciding if and when the changes should be implemented.
Version management	This involves **keeping track of the multiple versions of system** components and ensuring that changes made to components by different developers do not interfere with each other.
System building	This is **the process of assembling program components**, data, and libraries, and then compiling and linking these to create an executable system.
Release management	This involves **preparing software for external release** and keeping track of the system versions that have been released for customer use.

CONFIGURATION MANAGEMENT PLANNING

- Define **what is to be managed**, the configuration items, and a scheme to identify them (e.g. by their path in the file system)
- Determine **who is responsible** for the configuration management procedures
- Define the configuration management **policies** to be followed
- Specify the **tools** to be used for configuration management and the **processes** for using these tools
- Describe the **structure of the configuration database** that is used to record configuration information

TYPICAL QUERIES TO A CONFIGURATION DATABASE

- Which customers have taken delivery of a particular version of the system?
- ***What hardware and operating system is required to run a given system version?***
- How many versions of a system have been created and what were their creation dates?
- ***What versions of a system might be affected if a particular component is changed?***
- How many change requests are outstanding on a particular version?
- ***How many reported faults exist in a particular version?***

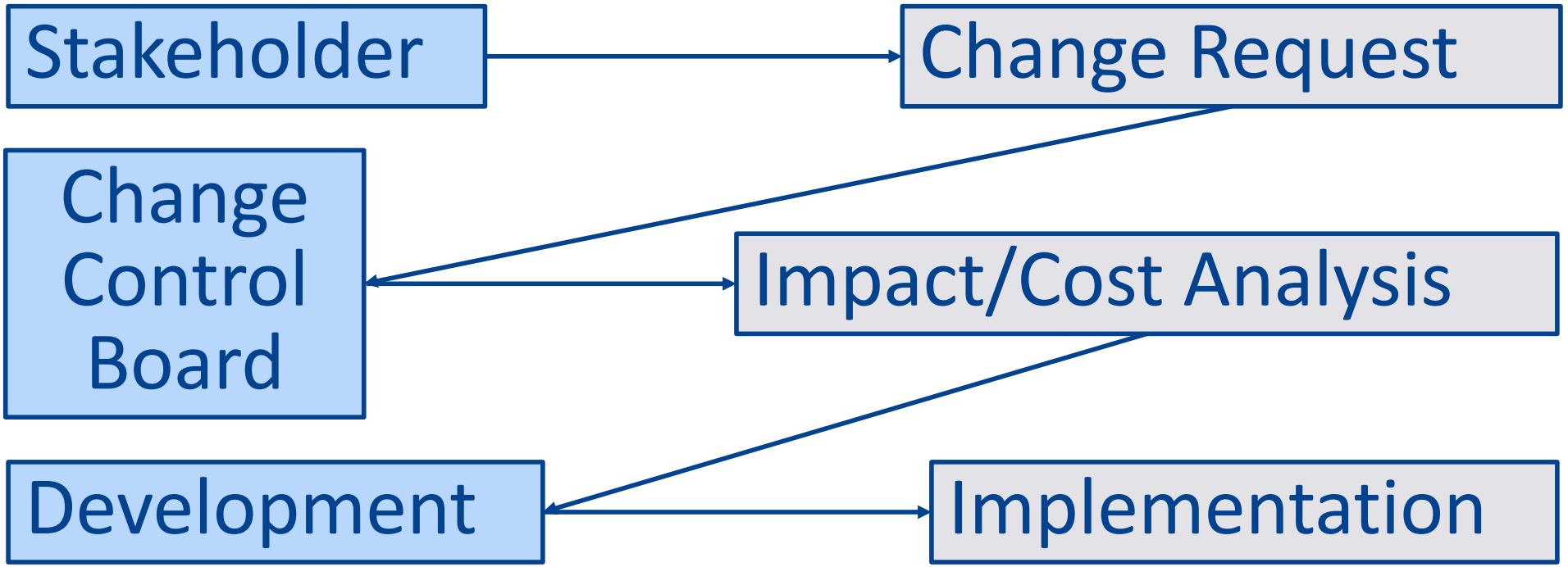
The configuration management plan determines what is stored in the database and what kind of queries are possible.

SOME TERMINOLOGY

Term	Explanation
Configuration item or software configuration item (SCI)	Anything associated with a software project (design, code, test data, document, etc.) that has been placed under configuration control. There are often different versions of a configuration item. Configuration items have a unique name.
Configuration control	The process of ensuring that versions of systems and components are recorded and maintained so that changes are managed and all versions of components are identified and stored for the lifetime of the system.
Version	An instance of a configuration item that differs, in some way, from other instances of that item. Versions always have a unique identifier.
Baseline	A baseline is a collection of component versions that make up a system. Baselines are controlled, which means that the versions of the components making up the system cannot be changed.
Release	A version of a system that has been released to customers (or other users in an organization) for use.

CHANGE MANAGEMENT

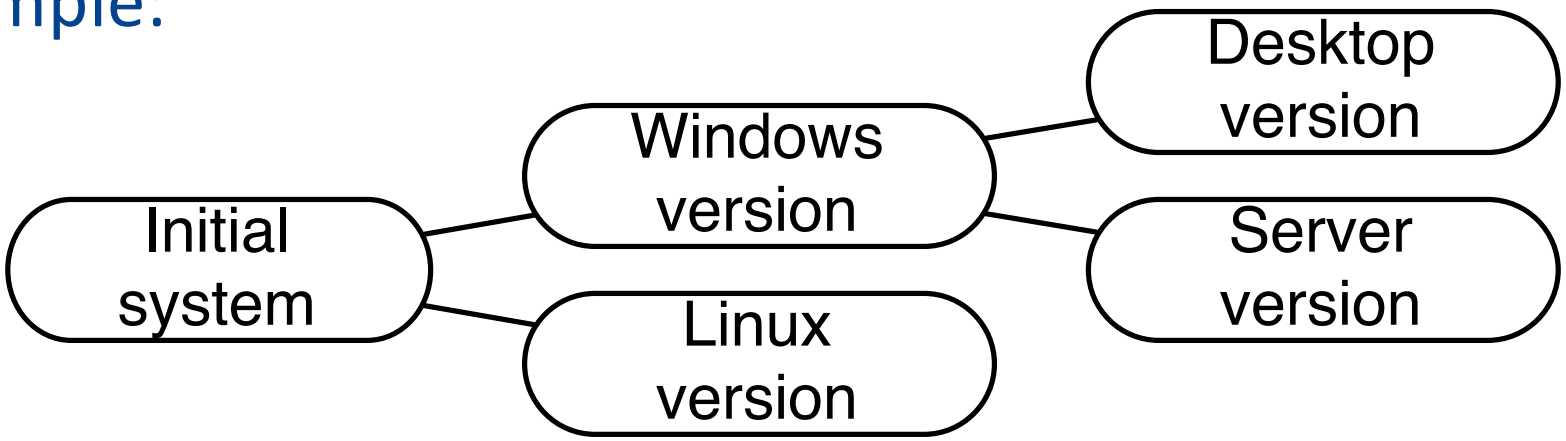
› Typical change management flow:



VERSION MANAGEMENT

- Software may be produced
  - for different computers
  - for different operating systems
  - for different customers with client-specific functions

- Example:



VERSION MANAGEMENT OBJECTIVE

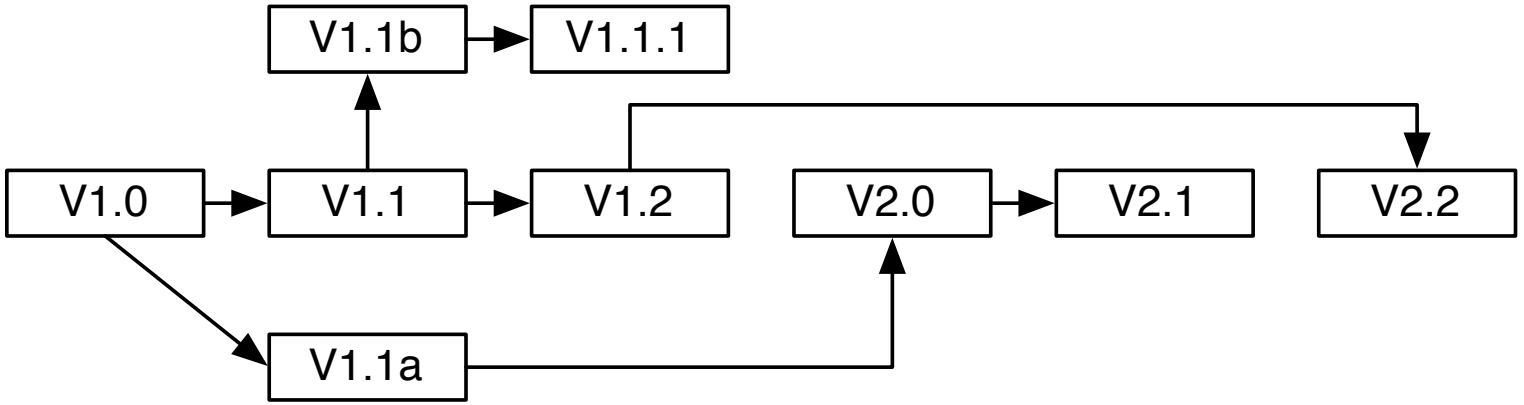
- **Objective** is to ensure that versions of a system
  - may be retrieved when required
  - are not accidentally changed by the development team

- **Functionality** of versions:
  - Enhanced performance
  - Repaired software faults
  - Targeted at different hardware or software platforms

- Versions with only small differences are also called **variants**

VERSION IDENTIFICATION

› Version numbering



› Attribute-based identification
  - › vi (platform = Linux)
  - › vi (platform = Windows)
  - › for “vi” software variants for Windows or Linux
  - › (what is ”vi”? https://en.wikipedia.org/wiki/Vi)

RELEASE MANAGEMENT

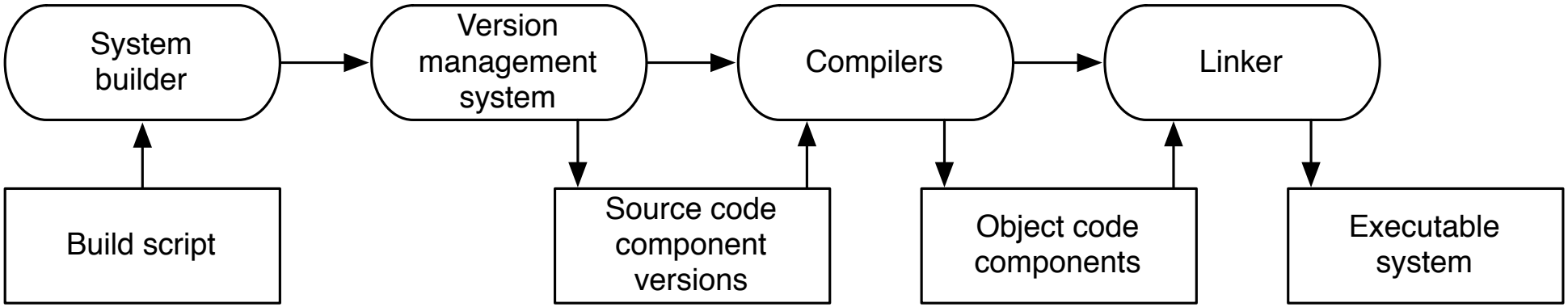
A release may contain, for instance,

- The executable software
- Configuration files (defining how the release should be configured for particluar installations)
- Datafiles (that are needed for successful system operation)
- An installation program (that is used to help install the system on target hardware)
- Electronic and paper documentation (describing the system)
- Packaging and associated publicity (that have been designed for that release)

FACTORS INFLUENCING THE RELEASE STRATEGY

Technical quality of the system	Serious system faults that have been reported may demand a fault repair release
Platform changes	A new version of an operating system may require a new release
Competition	A competing product may necessitate a new release
Customer change proposals	Customers may have made and paid for a specific set of system change proposals

BUILDING A SYSTEM



- All components in the appropriate version and all files must be available
- Building is usually an **automated process** (IDE, build script, build server)

EXAMPLES OF VERSION CONTROL SYSTEMS

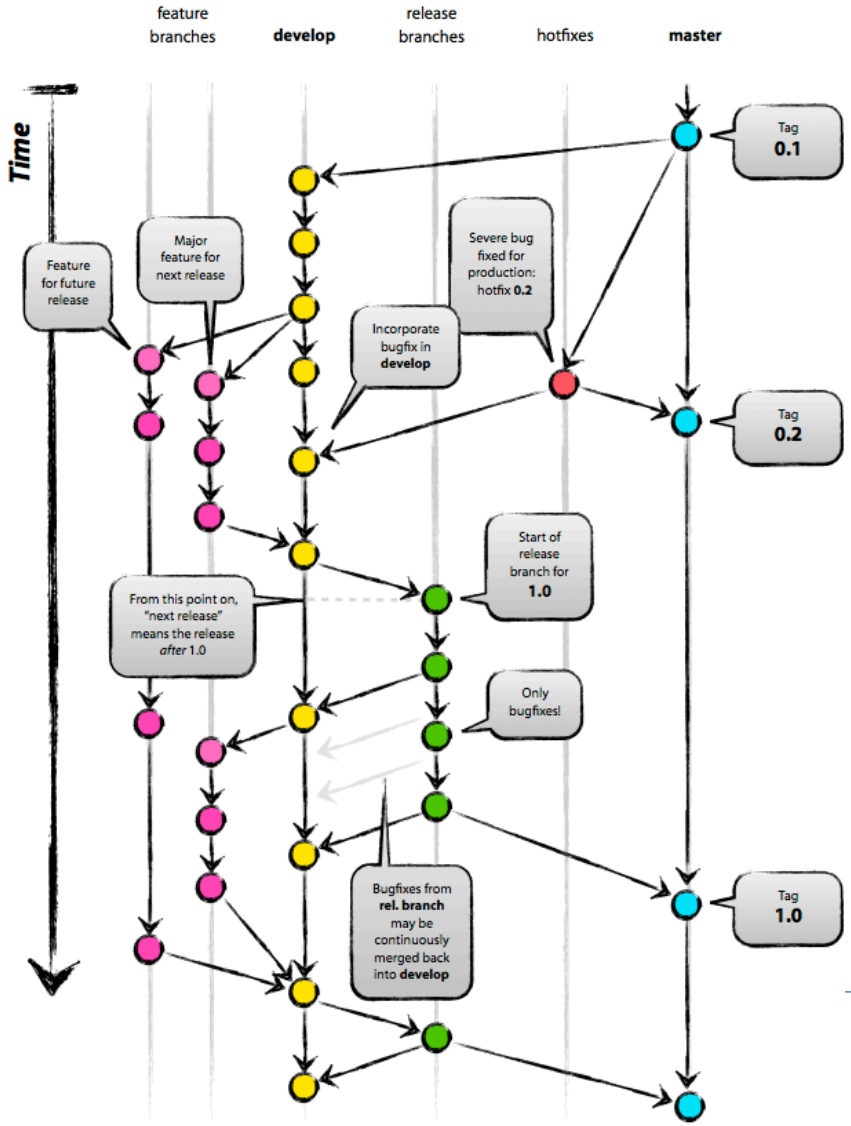
› https://en.wikipedia.org/wiki/Comparison_of_version_control_software

Version control terminology:

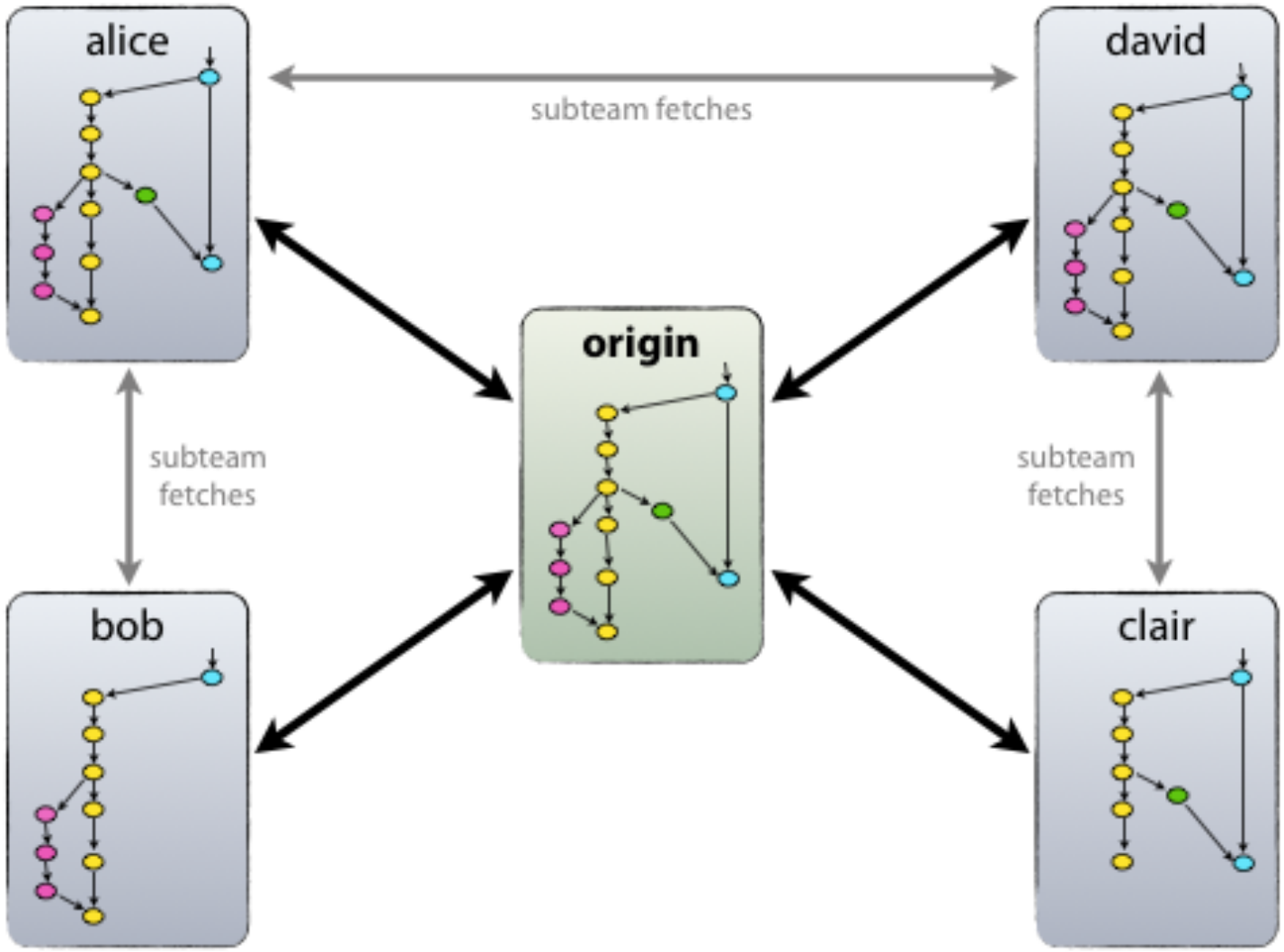
› https://en.wikipedia.org/wiki/Version_control

EXAMPLE (HTTP://NVIE.COM/POSTS/A-SUCCESSFUL-GIT-BRANCHING-MODEL/)

› A successful Git branching model (What is ”git”? https://git-scm.com)

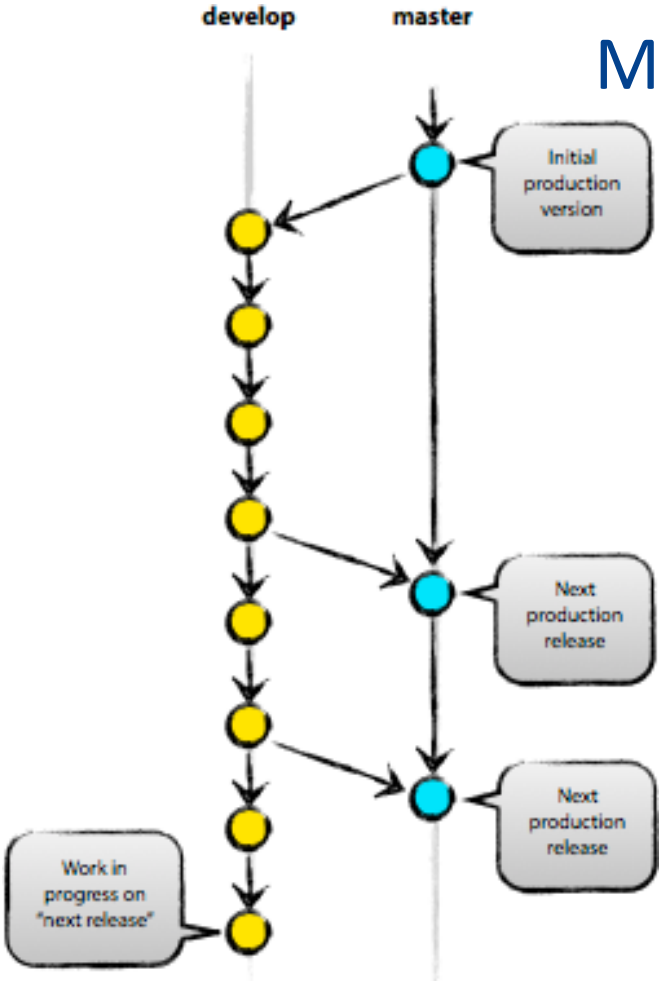


DISTRIBUTION OF REPOSITORIES

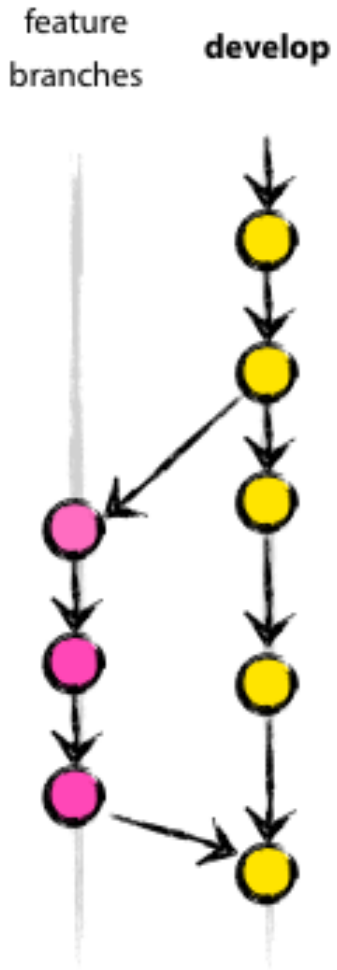


MAIN DEVELOPMENT BRANCHES AND FEATURE BRANCHES

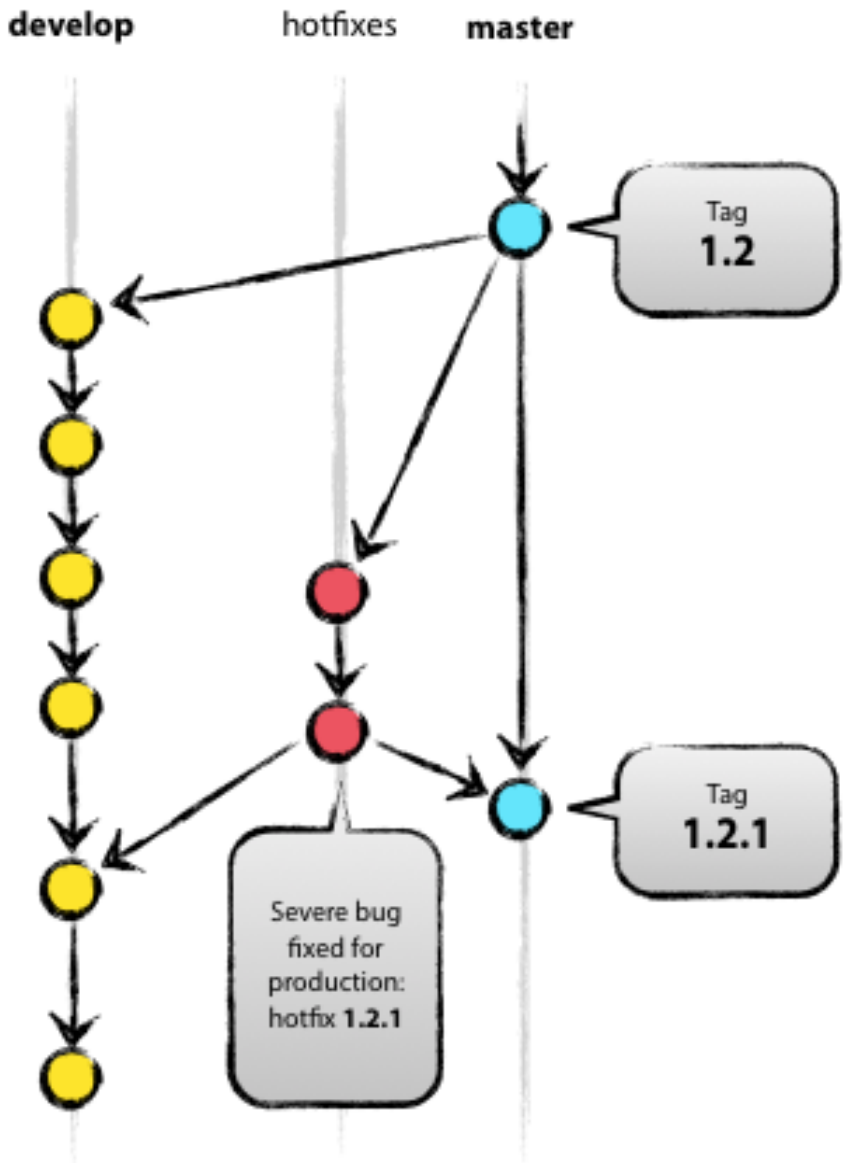
Main Branches



Feature Branches



HOTFIXES


develop
hotfixes
master
Tag
1.2
Tag
1.2.1
Severe bug
fixed for
production:
hotfix 1.2.1

CONCLUDING REMARKS

- Configuration management manages system change
- Requires a scheme for referring to documents under configuration control
- System releases include
  - Executable code
  - Data files
  - Configuration files
  - Documentation and more
- A good strategy for configuration management is important
- Good strategies are available