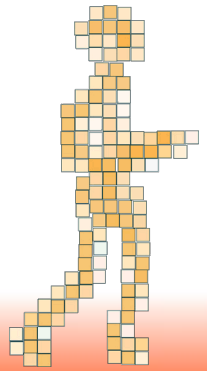
# Praqma & aragost Trifork

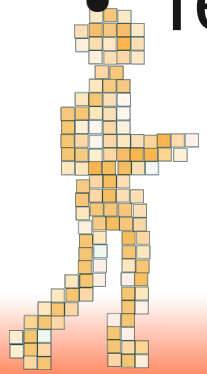
# Bridging *Mercurial* and *ClearCase* using *Jenkins*

Leif Sørensen & Christian Wolfgang
*Praqma, www.praqma.net*
Copenhagen, Denmark
les@praqma.net, chw@praqma.net
•
Klaus Bucka-Lassen
*aragost Trifork, www.aragost.com*
Zürich, Switzerland
kbl@aragost.com



# Agenda

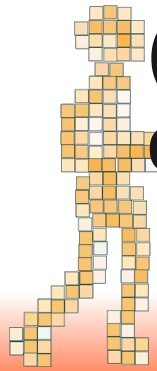
- Who We Are & Background
- CVCS vs. DVCS
- Migration Strategies
- Bridging & Flow
- Mapping ClearCase & Mercurial
- Demo
- Technical Implementation



PRAQMA

# History

- Became interested in DVCS in 2008 and have since developed several Open Source extensions and solutions
  - Matrix reloaded (Jenkins plugin)
  - ClearCase UCM (Jenkins plugin)
  - javahg (java api for mercurial)
  - onsub (run commands on *all* sub-repositories)
  - and about 20 others
- Help customers get set for the future (workflow, architecture, migration, support, etc.)

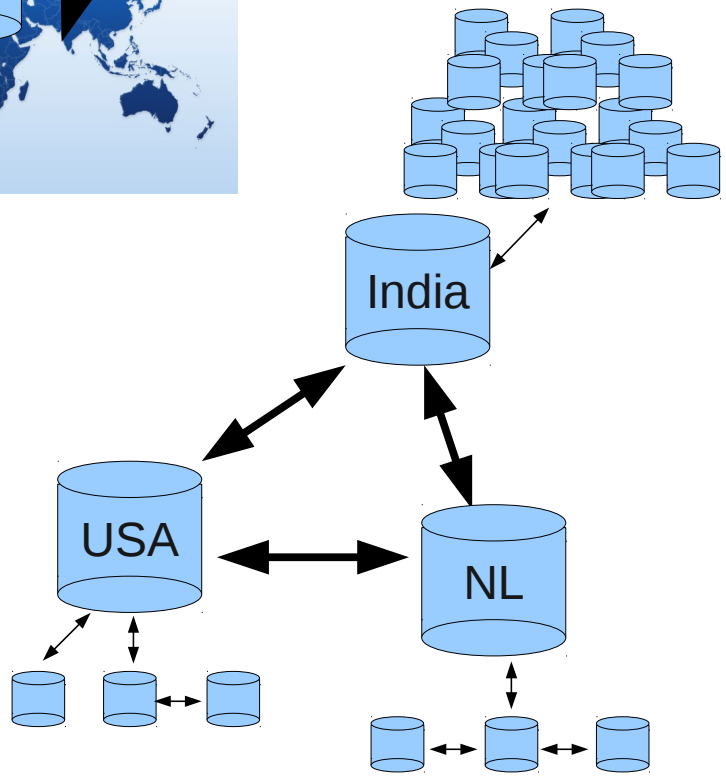


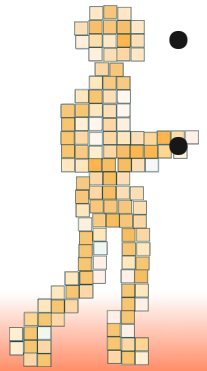
PRAQMA

# The Future is Distributed

- CVCS (Perforce, SVN, CVS, ClearCase, etc.)
  - Network **latency**
  - **Single point of failure**
  - **Constrained** workflow

- DVCS (Mercurial, Git, Bazaar, etc.)
  - **Flexible** (workflow)
  - **Distributed**, **off-line** operations (rich set of fast local operations)
  - New tool to **learn**
  - **Space** consumption

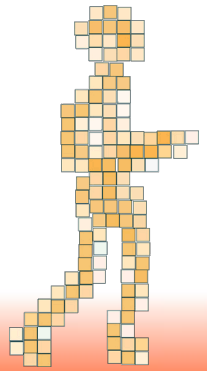


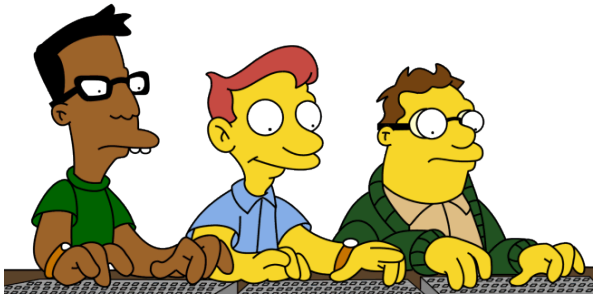



PRAQMA

# It Gets Even Better ...

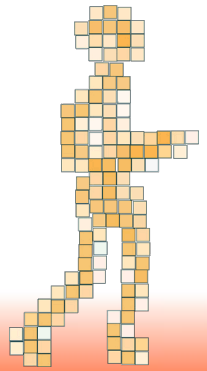
- Derived effects
  - Fine grained commits
  - Searchable history
  - Branching and merging becomes natural task
    (By the very nature of distributed work it has to!
    *“if something hurts, do it more often”*)
  - Better workflows emerge
    (e.g. feature branches; version maintenance; code reviews)





PRAQMA

# Management Convinced!?



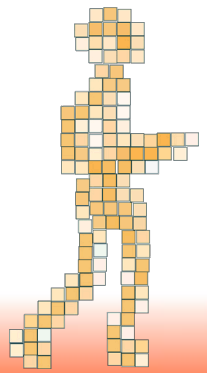
CVCS DVCS

# Traditional Migration Strategies

- Big Bang
  - Thorough analysis required
  - Some standstill for everybody
  - Bottleneck resources
    (consultants, training, support, etc.)
- Iterative / in stages (e.g. by project)
  - Interfacing between projects (sharing code)

High risk
& Expensive

Might not even
be an option



PRAQMA

# Too Expensive, Too Scary :-(

SVN

MKS

ClearCase

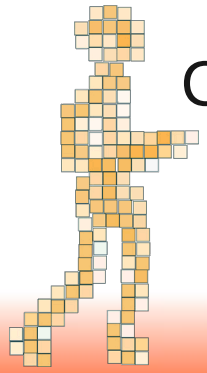
CVS

Perforce

Git

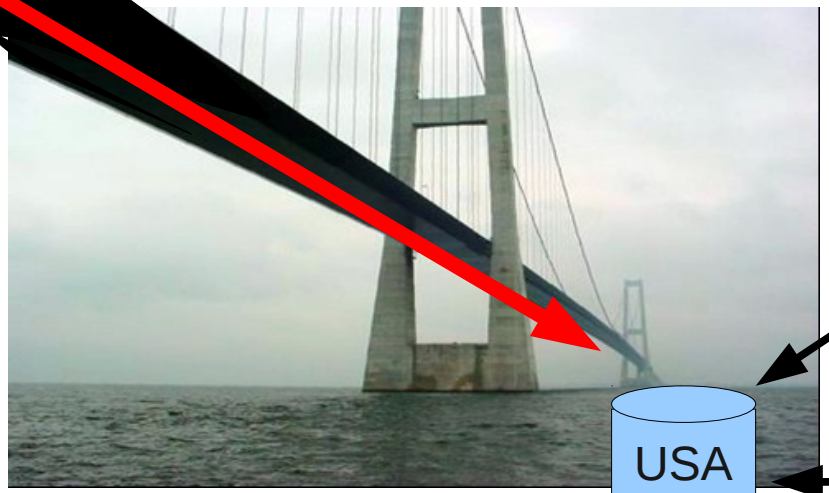
Bazaar

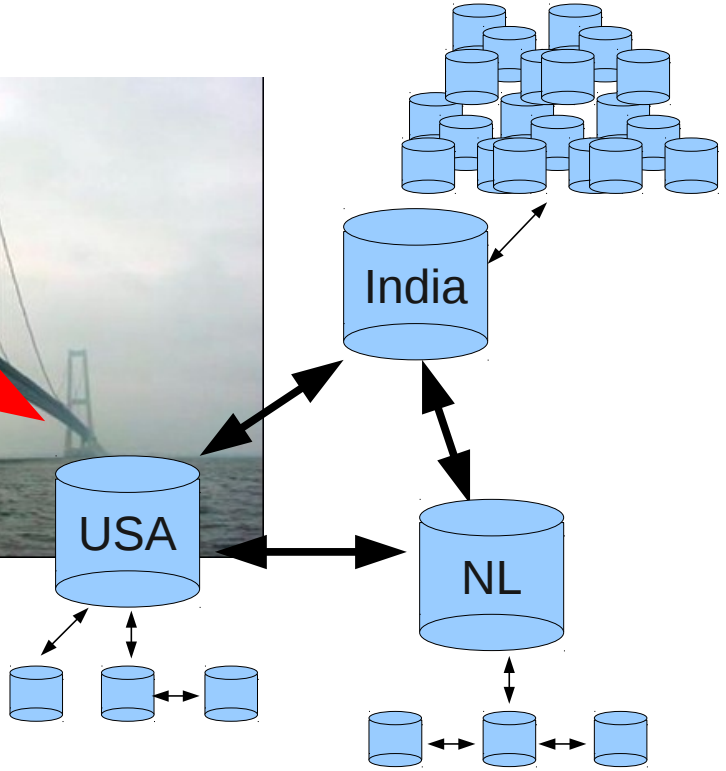
Mercurial

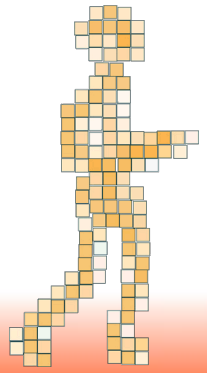


# Bridging

“Changes in one tool are automatically reflected in the other and vice versa”



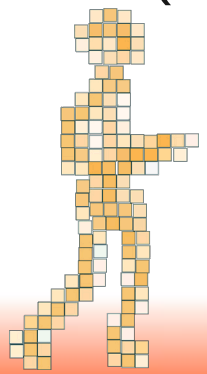


PRAQMA

# Bridging

- Get the best of two worlds
  (different tools for different tasks - happy developers, happy process people, happy managers)
- Protect your investment
- Defer commitment – Lean
  (Schedule irreversible decisions at the last responsible moment)
- Possible migration path
  (allows for iterative migration)

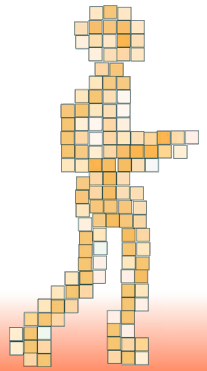


**Open Source**

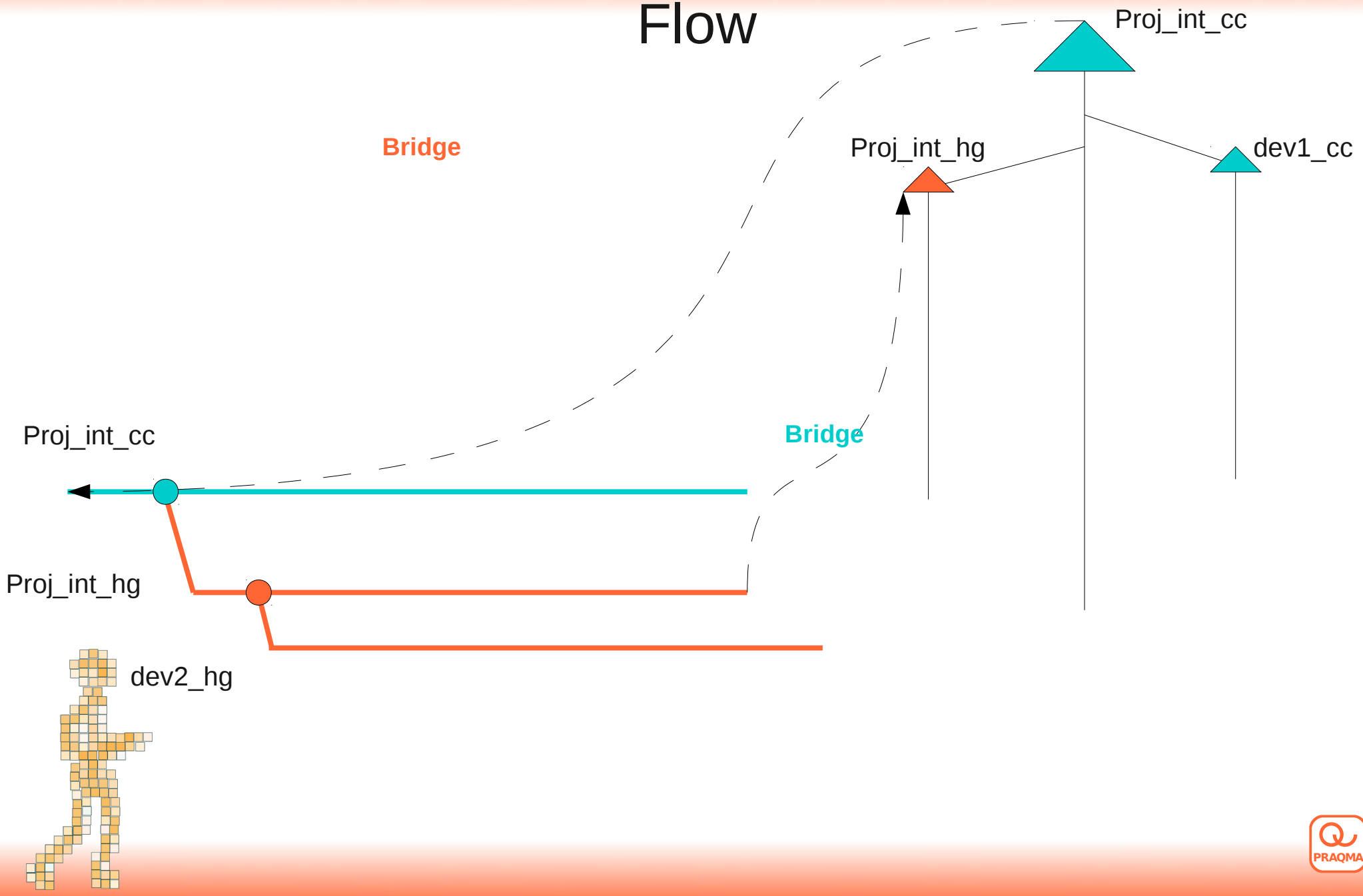
PRAQMA

# Praqma

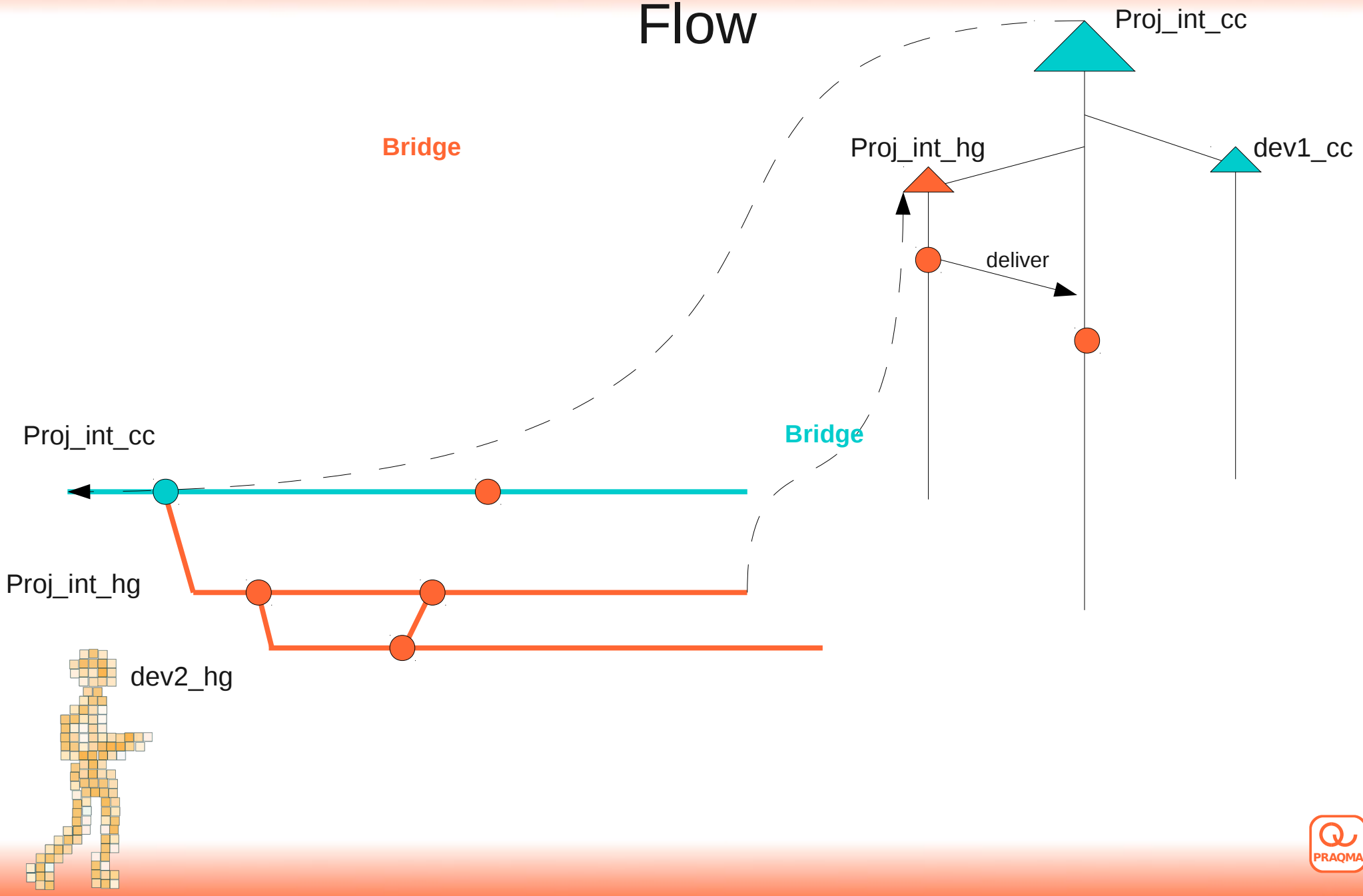
## Bridging Mercurial and ClearCase using Jenkins



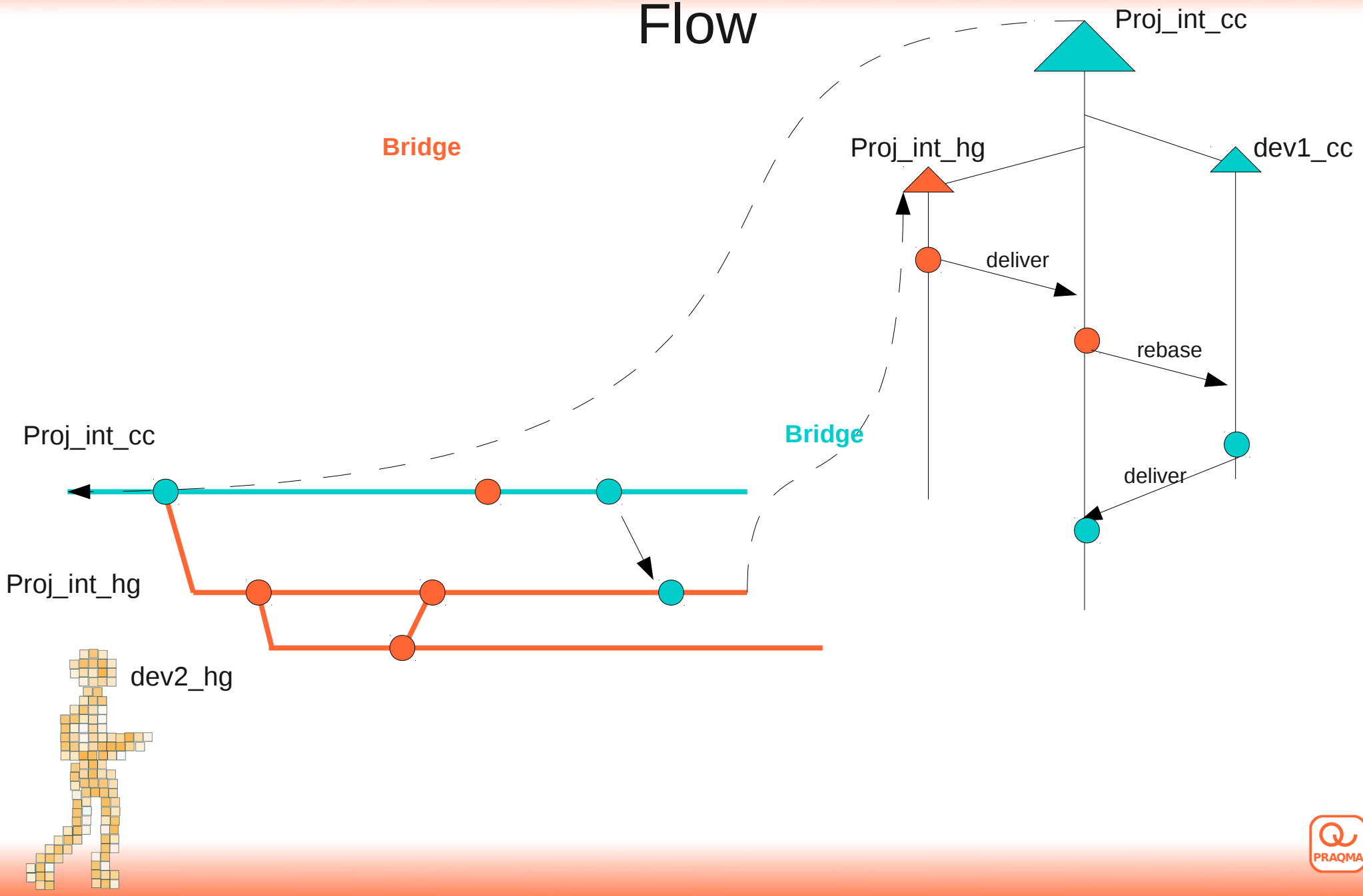
# Bridge
## Flow


Proj_int_cc
Bridge
Proj_int_hg
dev1_cc
Proj_int_cc
Bridge
Proj_int_hg
dev2_hg

# Bridge
## Flow

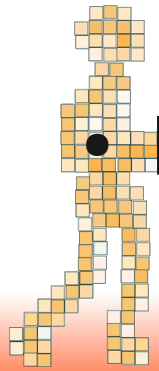

Proj_int_cc
Bridge
Proj_int_hg
dev1_cc
deliver
Bridge
Proj_int_cc
Proj_int_hg
dev2_hg

# Bridge
## Flow


Proj_int_cc
Bridge
Proj_int_hg
dev1_cc
deliver
rebase
deliver
Proj_int_cc
Bridge
Proj_int_hg
dev2_hg

# Bridging Version Control Systems

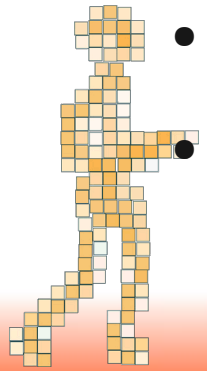
- Modern version control systems
  - Bases on the same conception
  - Almost trivial to bridge
  - Many bridges
- ClearCase UCM
  - Bases on different conception
  - Not trivial
  - No bridges
- Make ClearCase UCM behave



PRAQMA

# ClearCase and Mercurial

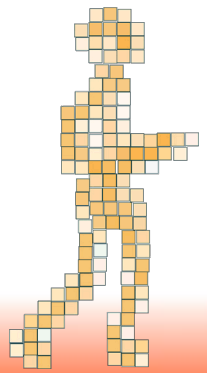
- Version control systems
  - ClearCase UCM
  - Mercurial
- Differences
  - ClearCase versions each element
  - Mercurial versions the repository state
- The need for a stable reference
  - Mercurial has commits
  - ClearCase has baselines



PRAQMA

# Demonstration

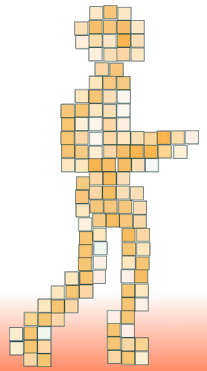
- Jenkins as scheduler
  - Using two Jenkins plugins
    - Mercurial plugin
    - ClearCase UCM plugin
  - Two different jobs
    - ClearCase as source, Mercurial as target
    - Mercurial as source, ClearCase as target



PRAQMA

# Jenkins - More than a build server

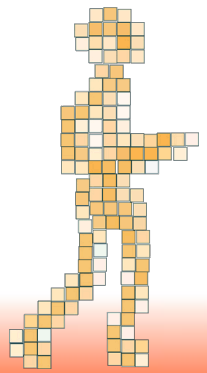
- Continuous integration server
- Open source, vivid community
- Many plugins, and more to come
- Implemented in Java



PRAQMA

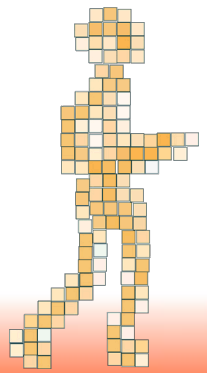
# Plugins, ClearCase UCM Plugin

- Developed by Praqma
- Three types of polling
  - Self, child and sibling
- Support for advanced work flows
- Baseline by baseline or latest
- Post build steps



# Plugins, Mercurial Plugin

- Get latest
- Does not support advanced work flows

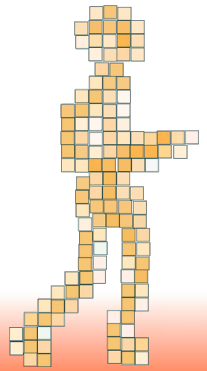


PRAQMA

# Project Information

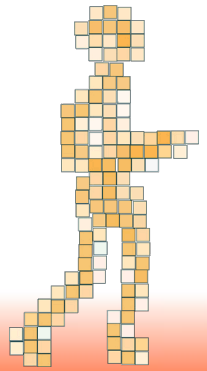
Abstract VCS Api:
https://github.com/Praqma/AbstractVcsApi

ClearCase UCM Plugin for Jenkins:
https://wiki.jenkins-ci.org/display/JENKINS/ClearCase+UCM+Plugin



PRAQMA

# Praqma & aragost Trifork

# Bridging *Mercurial* and *ClearCase* using *Jenkins*

Leif Sørensen & Christian Wolfgang
*Praqma, www.praqma.net*
Copenhagen, Denmark
les@praqma.net, chw@praqma.net
•
Klaus Bucka-Lassen
*aragost Trifork, www.aragost.com*
Zürich, Switzerland
kbl@aragost.com